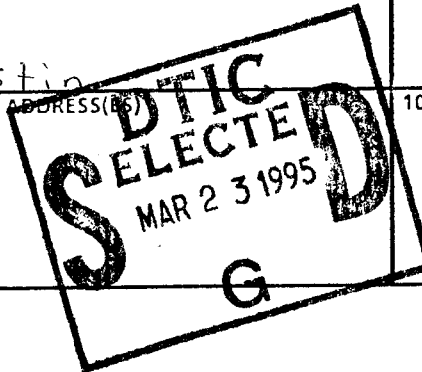


REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0185

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE May 95		3. REPORT TYPE AND DATES COVERED																					
4. TITLE AND SUBTITLE The Development of Methods & Techniques to detect & model the Underlying Structure of Chaotic systems				5. FUNDING NUMBERS																					
6. AUTHOR(S) James R. Bookhart Jr				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/CI/CIA																					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AFIT Students Attending: Univ of Texas Austin				95-005																					
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) DEPARTMENT OF THE AIR FORCE AFIT/CI 2950 P STREET, BDLG 125 WRIGHT-PATTERSON AFB OH 45433-7765				10. SPONSORING/MONITORING AGENCY REPORT NUMBER																					
11. SUPPLEMENTARY NOTES																									
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for Public Release IAW AFR 190-1 Distribution Unlimited BRIAN D. GAUTHIER, MSgt, USAF Chief Administration				12b. DISTRIBUTION CODE																					
13. ABSTRACT (Maximum 200 words)																									
<table border="1"> <tr> <td colspan="2">Accession For</td> </tr> <tr> <td>NTIS CRA&I</td> <td><input checked="" type="checkbox"/></td> </tr> <tr> <td>DTIC TAB</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Unannounced</td> <td><input type="checkbox"/></td> </tr> <tr> <td colspan="2">Justification _____</td> </tr> <tr> <td colspan="2">By _____</td> </tr> <tr> <td colspan="2">Distribution /</td> </tr> <tr> <td colspan="2">Availability Codes</td> </tr> <tr> <td>Dist</td> <td>Avail and/or Special</td> </tr> <tr> <td>A-1</td> <td></td> </tr> </table>						Accession For		NTIS CRA&I	<input checked="" type="checkbox"/>	DTIC TAB	<input type="checkbox"/>	Unannounced	<input type="checkbox"/>	Justification _____		By _____		Distribution /		Availability Codes		Dist	Avail and/or Special	A-1	
Accession For																									
NTIS CRA&I	<input checked="" type="checkbox"/>																								
DTIC TAB	<input type="checkbox"/>																								
Unannounced	<input type="checkbox"/>																								
Justification _____																									
By _____																									
Distribution /																									
Availability Codes																									
Dist	Avail and/or Special																								
A-1																									
14. SUBJECT TERMS				15. NUMBER OF PAGES 127																					
				16. PRICE CODE																					
17. SECURITY CLASSIFICATION OF REPORT		18. SECURITY CLASSIFICATION OF THIS PAGE		19. SECURITY CLASSIFICATION OF ABSTRACT																					
				20. LIMITATION OF ABSTRACT																					



19950322 133

98-216

The Development of Methods and Techniques
to Detect and Model the Underlying Structure
of Chaotic Systems

by

James Robert Bookhart, Jr., B.S.

Thesis

Presented to the Faculty of the Graduate School
of The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Master of Arts

The University of Texas at Austin

May, 1995

The Development of Methods and Techniques
to Detect and Model the Underlying Structure
of Chaotic Systems

APPROVED BY

SUPERVISING COMMITTEE:

Alvin Natta

William G. Less

Table of Contents

Chapter One: Introduction	1
Chapter Two: Definitions and Examples	9
Chapter Three: Methods	40
Chapter Four: Application	56
Chapter Five: Summary	102
Appendix A: Hurst Coefficient Program	108
Appendix B: Correlation Dimension Program	114
Appendix C: Lyapunov Exponent Program	122
Bibliography	125
Vita	128

CHAPTER ONE

INTRODUCTION

Linearity is a beautiful thing in science and mathematics. Frequently, it is easy to recognize and carries with it a long list of wonderful properties. Like many things we choose to describe as beautiful, linearity, too, is relatively rare in nature.

Until recently, science's approach to nonlinear systems has been to coerce the ill behaved system into a more friendly linear form. Some systems, however, stubbornly resist such linearization. Such a system, once linearized, may no longer exhibit the same behavior as its parent system. Granted, studying the new system is certainly much less painful but any answers found are to a completely different question.

Is all hope lost once the system being studied is found to be such a system? For many scientists and for many years the answer was "Yes!" Upon this discovery, the researcher would throw up his hands in disgust and look for a new model that was not so poorly behaved, hoping to find a rationalization for the nonlinear factor being deemed "insignificant anyway".

A recently renewed interest in nonlinear dynamics is causing a shift in this attitude. New ways of characterizing and describing these systems are being developed which show that all hope is not lost. Students of nonlinear dynamics are realizing that seemingly harmless nonlinear systems can exhibit monstrous behavior. This discovery sparks the hope that empirical data that appears equally monstrous may in fact have a simple underlying mechanism. These "monsters" live in a world called Chaos and this thesis describes and develops techniques for looking at these systems.

Definitions and Examples

Traditionally, systems have been labeled either deterministic or stochastic. Those systems that are predictable are called deterministic. The motion of celestial bodies is one example of a deterministic system. For hundreds of years, astronomers have been able to measure the position and velocity of heavenly bodies and, using planetary laws of motion, predict occurrences such as eclipses many years into the future. Systems that resist prediction methods other than probability models are stochastic and are

sometimes called random. As an example of a stochastic system, consider the number of customers to patronize a business on a given day. It may be possible to give general trends such as "business is always better toward the end of the month" or that "usually we have about 1,000 customers in a day" but predicting the actual number of customers is not possible. For these systems, the best predictions available utilize probability models.

Current research indicates there is middle ground that contains systems which are neither predictable nor random. These systems that appear to be random but are actually deterministic are called chaotic. Briefly, a system is chaotic if nearly identical initial conditions move far apart as they are iterated. This criteria is called local divergence or sensitive dependence on initial conditions. This condition distinguishes a predictable deterministic process from a chaotic process since the sensitive dependence leads to unpredictability. More precise definitions of chaos are given in Chapter 2.

The weather is an example of a chaotic system. In this context, the local divergence property is

frequently described as the "butterfly effect". The butterfly effect theory hypothesizes that the air currents created today by the flapping of a butterfly's wings do not simply "die out", rather they affect surrounding air currents which in turn affect still other air currents and so on. This would mean that the weather today may have been changed by the flight of a butterfly many years ago. If this is true, long term forecasting of weather is impossible even if a simple physical model can be built.

While the study of chaotic systems which lie in the middle ground between randomness and predictability is centuries old, the increased interest is relatively recent; three decades old, or so. This interest is sparked by the creation of random looking data sets from simple mathematical models and also by the development of computer resources which make possible the data intensive computational methods of studying these data sets. A classic example of such a data set comes from the so called "logistic equation" $x_{t+1} = \lambda \cdot x_t(1-x_t)$. For small values of the parameter λ , the system converges to either a fixed point or a cycle. As this parameter

is increased toward 4.0, all stable fixed points and cycles disappear and the data appear to be random values. Perhaps then, some seemingly random series of data are the result of similarly simple systems.

Contents of this analysis

Work in this analysis begins by generating data sets from at least three models known to exhibit chaotic behavior. The data sets are then treated as though the generating functions are unknown and several tests for randomness are applied to determine if the data may be the result of a chaotic system. Next, a model is built in an attempt to recreate the original model from the data. In addition, a statistical model is built using standard multiple regression and time series techniques. The traditional method uses a Box-Jenkins Autoregressive Integrated Moving Average (ARIMA) model [1, p.449]. The predictive power of the two models, chaotic and stochastic, is then compared to determine the effectiveness of the chaos approach.

Data sets studied in this analysis are from the Logistic equation, the Henon equations, and the Rossler equations.

Organization of chapters

A brief description of the concepts and definitions essential to the study of chaotic dynamical system as well as references for further reading are found in Chapter 2. Also in Chapter 2 are descriptions of the specific analysis tools that are applied in this thesis along with references.

The method of analysis is described in Chapter 3 and consists of three sections. The first section focuses on current techniques available to test for randomness in a data set being studied. Topics covered in this section include phase space analysis using the method of delay coordinates, the Hurst coefficient and Rescaled Range (R/S) analysis, the correlation integral and correlation dimension, the Lyapunov exponent, and the Shuffling diagnostic. The second section offers methods for modeling data that the tests for randomness imply may be chaotic using dynamical systems. The second section also outlines the use of the correlation dimension, phase portrait analysis and autocorrelations to choose appropriate model equations. These models are then fit to the data using a least squares regression and the

regression output is analyzed for suggestion of a more appropriate model. The third section describes analyzing the predictive power of the chaotic models and comparisons of this predictive ability with that of models built using classical time series techniques. In this section, the system will be iterated using the deterministic model and also a classic time series model and the number of iterations before significant deviation from the original data set computed for both. The results confirm that the data set can be better modeled using the methods of chaos.

These steps are applied to the generated data sets and a description of the analysis is given in Chapter 4.

Finally, Chapter 5 provides a summary of the results along with conclusions drawn from these results. Also included in this chapter are recommendations for further study and suggested areas for extension of this work.

Objective

The goal of this paper is to demonstrate the process outlined above which combines several current

techniques and to outline a procedure for testing and modeling any series of data suspected of having an underlying deterministic chaotic explanation.

CHAPTER TWO

DEFINITIONS AND EXAMPLES

This chapter provides the basic concepts and definitions essential to studying things that live in the world called Chaos.

Time Series

Throughout this paper the term **series** refers to a list of data taken at discrete intervals and arranged according to the order in which the observation was made. For example, the monthly Dow Jones Industrial Average would be called a series $\{x_t\}$ where $0 \leq t \leq N-1$. The number N represents the number of months in the list of data to be studied. The term x_0 would be the Dow Jones average for the first month and x_i represents the average for the i^{th} month. The choice of starting index, in this case zero, has no effect on the behavior of the system. Starting with zero is the most common convention because it allows the interpretation of the index as the observation taken t units after the initial observation. If, as in the case of the Dow Jones average, the index variable is time, the series is called a **time series**.

The use of series in this context is the same as that adopted by the current literature, which is quite different from the mathematical use involving summation. Strictly speaking, the lists used in this study are sequences of observations.

Dynamical Systems

The **state** of a system is the condition of the system for a specified value of the index variable. For the Dow Jones average, the state of the system is the condition of all factors that impact the market. The time series of Dow Jones averages is a one dimensional observable which is a representation of the state of the system. The **state space** of a system is where these states live. For the Dow Jones average, the state space studied is one-dimensional, the positive real line. Other systems may have higher dimensional state spaces. For example, one may wish to simultaneously study the Dow Jones and the S&P 500 averages. This state space would be two dimensional since it takes the value of both averages to specify the state of the system.

The Dow Jones average is an example of a time series originating from observations of some "real

world" system, time series can also be created by taking values of some mathematical system at discrete time intervals. Dynamical systems are common sources of such data. A good, intuitive explanation of what is meant by a **dynamical system** is given in [2, p.2]: "A dynamical system can be thought of as any set of equations giving the evolution of the state of a system from a knowledge of its previous history". Simply put, applying the knowledge of where we have been and iterating with the dynamical system tells us where we are going.

Fractal Geometry and Attractors

The time series generated from a dynamical system is called a **trajectory**. In a sense, the trajectory is the path the system follows as it moves through its state space. A subset of the state space toward which almost all sufficiently close trajectories are attracted is called an **attractor** [3, p.73]. An attractor that has a fractal structure is called a **strange attractor**. The most common characteristic of a structure which is called fractal is a non-integer dimension. This fractal dimension is always no greater than the (Euclidean) dimension of the space in

which it is embedded. Since a strange attractor is a common, but not necessary, characteristic of a chaotic system, fractal geometry is frequently encountered in the study of chaos.

Several different measures of fractal dimension exist, but probably the most common is called the box counting dimension [4, p.34]. This dimension is defined by covering the image with n -dimensional boxes whose sides have length ϵ where n is the dimension of the embedding space. The number of boxes required to cover the image is then counted. This process continues for smaller and smaller boxes. The following equation, in the limit, defines the dimension:

$$D = \lim_{\epsilon \rightarrow 0} \sup \log N(\epsilon) / \log(1/\epsilon)$$

D = fractal dimension

$N(\epsilon)$ = Number of boxes to cover image

ϵ = length of sides of boxes

For further reading about fractal geometry, [5] is an excellent reference.

Attractors can normally be found only if the equations for the system being studied are known and even then finding the attractor is not guaranteed. In

studying real systems using dynamical systems as models, techniques are employed to reconstruct the attractor from the (finite) data available. Although the reconstructed attractor is only an approximation, it may hold useful information about the dynamics of the system and provide clues to the best ways to model it.

Deterministic, Stochastic and Chaotic

The requirement for a system to be called **deterministic** is just that it not be stochastic. A process is called **stochastic** if one or more variables are random. A variable is said to be **random** if it takes on values according to some (perhaps unknown) probability distribution. Although the definition of chaos varies from reference to reference, there are some basic requirements common to most definitions: (1) sensitive dependence on initial conditions , (2) some degree of regularity and (3) recurrence. Here, a point is called **recurrent** [6, p. 48] if for any open interval about that point, the system returns to that interval after a finite number of iterations. Most chaotic systems are low dimensional, non-linear, deterministic dynamical systems. Although completely

deterministic, the non-linearities cause these systems to appear stochastic. With the basic definition of a chaotic dynamical system completed, the focus now shifts toward ways to look at the behavior of the system and tools used to understand this behavior.

Graphical Analysis Tools

There are several graphical views of a time series that are frequently enlightening. These plots are the first diagnostic tools applied in attempting to determine if some deterministic structure exists. The first, the **time plot**, is a graph of the series values as the dependent and the index variable as the independent variable. The name "time plot" may seem misleading since the index variable is not necessarily time. Since most series have time as the index, however, this name is tolerable and is used throughout the paper. The **phase portrait** plots one series value against previous values. For example, if the time series is one dimensional $\{x_t\}$, one example of a (two dimensional) phase portrait would graph x_{t+1} versus x_t . The phase portrait may also be higher dimensional. For example, a plot of x_{t+2} on the z-axis, x_{t+1} on the y-axis, and x_t on the x-axis of \mathbb{R}^3

is a three-dimensional phase portrait. The two-dimensional example uses a **lag period** of one. The phase portrait for a lag period of n plots x_{t+n} against x_t . Building a phase portrait in this way is also called the method of delay coordinates for phase space reconstruction.

A **web diagram** (Figure 2.1 below) is a variation on this idea in two dimensions [4, p.23]. Here the pairs (x_t, x_{t+1}) are plotted with the pairs (x_t, x_t) . Then a trajectory is traced alternately connecting (x_t, x_t) to (x_t, x_{t+1}) and then (x_t, x_{t+1}) to (x_{t+1}, x_{t+1}) . To better understand the web diagram, consider a data set generated from a known function. Plot this function as well as the line $x_t = x_{t+1}$ and begin at some starting point, x_0 . Then, draw a line vertically to the curve representing the function followed by a horizontal line to the line $x_{t+1} = x_t$ and so on. This representation shows the orbits of the function f defined by $f(x_t) = x_{t+1}$ whereas the phase portrait shows only the points themselves. Often this gives a better "feel" for the dynamics of the system than a simple phase portrait. Caution should be used when working with long lists of data for which the function is not

known, as the web diagram may tend to hide some of the structure with unnecessary clutter.

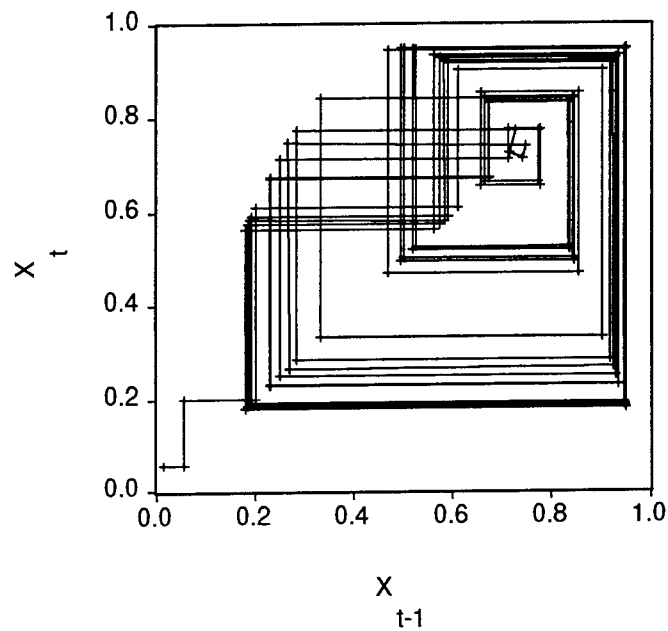


Figure 2.1 Web diagram for logistic equation

Hurst Coefficient

The **Hurst coefficient**, H , is the first diagnostic tool applied in this analysis ([7, Chap.4],[8, p.386],[9]). First, an intuitive explanation of this coefficient is given and then the details of calculating its value are provided.

To apply general theory to a specific time series $\{x_t\}$ with $0 \leq t \leq n$, the first step is to transform the given data set to a new set $\{x'_t\}$ which has mean zero and standard deviation one. This is easily enough done by subtracting the mean value from each value in the series and then dividing by the sample standard deviation.

Analysis using the Hurst coefficient relies on the fact that if the data set being studied is the result of a random process, then the present value of the series is not influenced by previous values. For such a process, the series $\{\sigma_m\}$ defined below becomes what is known as a random walk.

$$\sigma_m = \sum_{t=1}^m (x'_t) \quad \text{for } 1 \leq m \leq n$$

For a process to be called a random walk, it must move from the current state according to some set of rules which do not depend on the current state. As an example of a random walk, consider a discrete time sequence of values on the real line which begins at zero and moves at each point in time with equal

probability one unit in either the positive or negative direction.

The range, R_n , of this new series, $\{\sigma_m\}$ is defined as:

$$R_n = \max_m (\sigma_m) - \min_m (\sigma_m) \quad 1 \leq m \leq n$$

When the data is the result of a random process, it is a well known result that this range scales as the square root of n , the length of the series. That is, the limit as n tends to infinity of $R_n/n^{0.5}$ exists and is finite and positive.

For a data set which is not the result of a random process, the present value is influenced by previous values. This is what is referred to as a memory effect. In this case, the series $\{\sigma_m\}$ is no longer a random walk. For such a series, the range scales as some power of n other than 0.5.

The scaling exponent described is what is known as the Hurst coefficient. The conditional distribution of the current value given previous values in the series is, at best, very difficult to estimate while the Hurst coefficient gives

qualitatively similar information and is relatively simple to estimate.

If an infinite amount of data were available, it would be possible to test whether the sequence $(R/S)_n / n^{0.5}$ converges. Since only a finite amount of data is available, an alternative form of the scaling law must be applied. The form of the law being applied states that for a random walk, the expected value of $(R/S)_n$ is given by $c \cdot n^{0.5}$. The method used to estimate the Hurst coefficient described below averages the values of $(R/S)_n$ for disjoint subseries for each value of n starting with 3 and continuing to $N/2$ for a series of length N . This "sample mean" approximates the expected value of $(R/S)_n$ for each n and is an effective use of limited data when estimating H .

The method for estimating the Hurst coefficient for a series $\{x_t\}$, with $0 \leq t \leq N-1$, begins by partitioning the time period $0 \leq t \leq N-1$ subintervals of length n . The subintervals are labeled I_a with $1 \leq a \leq A$ with A equal to the greatest integer less than N/n . The first subinterval is from 0 to $n-1$ and each subinterval is from $k \cdot a - 1$ to $(k+1) \cdot a - 1$ for some

integer k . The later points in the original interval remaining after partitioning are discarded. Each member of the original series, x_t , is then renamed $N_{k,a}$ with $1 \leq k \leq n$. In the new labeling system, k represents the position within the subinterval and a represents the subinterval. For each subinterval, the mean value of the series values, μ_a , is calculated. The cumulative departure from the mean is given by the formula:

$$X_{m,a} = \sum_{k=1}^m (N_{k,a} - \mu_a) \quad 1 \leq m \leq n$$

The range for each subperiod is defined as the difference between the maximum and minimum of $X_{m,a}$ according to:

$$R_{I_a} = \max_m (X_{m,a}) - \min_m (X_{m,a}) \quad 1 \leq m \leq n$$

The sample standard deviation is calculated for each subperiod according to:

$$S_{I_a} = \left((1/n) \cdot \sum_{k=1}^n (N_{k,a} - \mu_a)^2 \right)^{0.5}$$

Each range is now normalized using the following formula:

$$(R/S)_n = (1/A) \cdot \sum_{a=1}^A (R_{I_a}/S_{I_a})$$

This is repeated for each integer value of n less than $(N-1)/2$ beginning with $n = 3$. Finally, the Hurst coefficient, H , is given by the equation:

$$(R/S)_n = (c_1 \cdot n)^H$$

In the limiting case of an infinite amount of noise-free data, this equation holds true for all n . Using a finite amount of data, H must be estimated using a least-squares regression. To approximate the value of H , logarithms are first taken:

$$\log(R/S)_n = c_2 + H \cdot \log(n)$$

The value of H is then the β_1 coefficient in the regression equation:

$$\log(R/S)_n = \beta_0 + \beta_1 \cdot \log(n)$$

This calculation is now demonstrated for two simple examples. For the first example, consider the system $\{x_t\}$ $0 \leq t \leq 10$ defined by:

$$x_{t+1} = x_t + 1 \quad \text{with } x_0 = 0$$

For this example, possible subintervals are of lengths three, four, and five. The values of the sub-series after subtracting the subinterval mean are:

$$\begin{aligned} n = 3 & \quad (-1, 0, 1) \\ n = 4 & \quad (-1.5, -0.5, 0.5, 1.5) \\ n = 5 & \quad (-2, -1, 0, 1, 2) \end{aligned}$$

The cumulative departure from the mean for each subinterval is:

$$\begin{aligned} n = 3 & \quad (-1, -1, 0) \\ n = 4 & \quad (-1.5, -2, -1.5, 0) \\ n = 5 & \quad (-2, -3, -3, -2, 0) \end{aligned}$$

The range and sample standard deviation for each subinterval are:

$$\begin{aligned} n = 3 & \quad \text{range} = 1.0 & \quad \text{s.d.} = 0.816 \\ n = 4 & \quad \text{range} = 2.0 & \quad \text{s.d.} = 1.118 \\ n = 5 & \quad \text{range} = 3.0 & \quad \text{s.d.} = 1.414 \end{aligned}$$

The rescaled range $(R/S)_n$ for each value of n is:

$$(R/S)_3 = 1.225$$

$$(R/S)_4 = 1.789$$

$$(R/S)_5 = 2.121$$

Taking logarithms and using the β_1 coefficient in the following regression equation to estimate the Hurst coefficient yields the result $H = 1.08$.

$$\log(R/S)_n = \beta_0 + \beta_1 \cdot \log(n)$$

This coefficient value should be 1.0 and the larger approximation is the result of applying a method which increases in accuracy with sample size to a very small data set. Calculation of the Hurst coefficient for the series same series, that is $x_{t+1} = x_t + 1$, using 100 data points yields $H=1.02$ and for 1,000 points $H=1.005$ supporting the conclusion that the true value of H is 1.00. The actual results are not significant to the work done here, the objective is simply to illustrate the process of estimating the Hurst coefficient for a simple data set.

Repeating this calculation for a strictly alternating series, that is $\{x_t\}=\{0,1,0,1,0,1, \dots\}$,

produces an interesting result. The plot of $\log(R/S)_n$ against $\log(n)$ produces two lines, one for even values of n and the other for odd values (Figure 2.2). Both lines have slope which goes to zero as n increases showing that the scaling of $(R/S)_n$ is not proportional to N , that is $H=0.0$. Apparently the appearance of two lines is because the mean for subseries whose length is an even number is 0.5, and when the length is odd the mean is not 0.5 and therefore the series is not

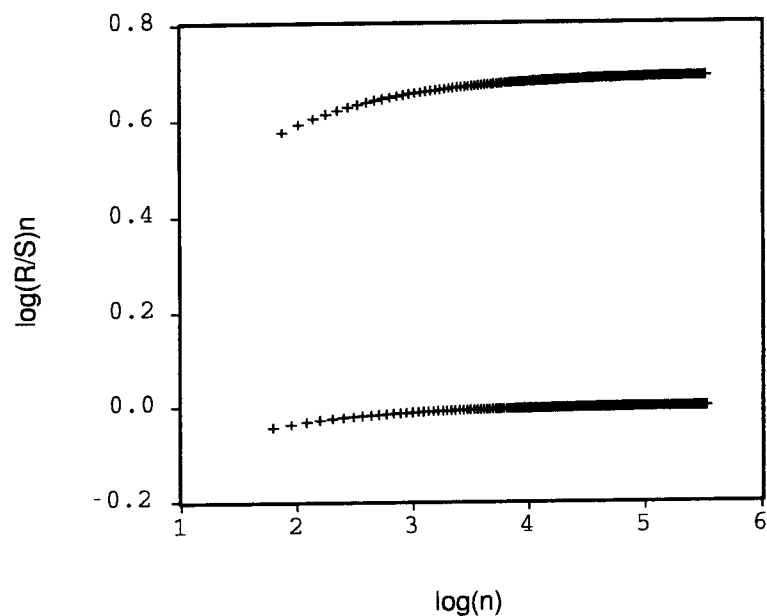


Figure 2.2 R/S plot for alternating series

symmetrically distributed about the mean. The full explanation and impact of this result is not completely understood but this does not seem to be a barrier to the application of R/S analysis in this thesis.

The actual coefficient values range from 0 to 1.0 [7, p.61]. Although the earlier example demonstrates that, particularly for small values of n , the estimated coefficient may not necessarily be in this range. For an independent and identically distributed random variable, the value of H tends asymptotically to 0.5 as n tends to infinity due to the previously mentioned scaling law for random walks. A value of the Hurst coefficient significantly different from 0.5, therefore, provides evidence that there may be a deterministic structure. Note that it does not guarantee the existence of structure, rather it gives no evidence showing the process is random.

The values of H which are greater than 0.5 are called persistent. **Persistent** series have, at each point in time, a higher probability of following the current trend than reversing it. If the Dow Jones average were shown to be persistent, then if this

month's average is higher than the previous month's, next month's average has a greater probability of being higher than lower. A Hurst coefficient less than 0.5 indicates the system is antipersistent. For **antipersistent** series, the probabilities are reversed. Sometimes the term "mean reverting" is used in place of antipersistent in the case of the existence of a stable mean and "trend reinforcing" is used for persistent.

When using rescaled range analysis for small data sets such as the ones in this thesis, Peters cautions that the estimated H values will generally be higher than the true value. To help deal with this problem, Peters offers an equation which estimates the expected value of $(R/S)_n$ as a function of n [7, p.71]. This estimate is the result of calculating the average of $(R/S)_n$ values for a large number of sets of n random numbers generated from Monte Carlo simulations. Peters calculated the value of H for a wide range of values of n and the following equation was derived:

$$E((R/S)_n) = ((n-0.5)/n) \cdot (n \cdot \pi/2)^{-0.50} \cdot \sum_{r=1}^{n-1} ((n-r)/r)^{0.5}$$

For this analysis, this equation was used to generate $(R/S)_n$ values between 3 and 500 (those which will be used when analyzing 1,000 point data sets) and the Hurst coefficient estimated. The result is an expected H value of 0.558 for a random data set consisting of 1,000 points.

To determine which values of H should be considered significantly different from a given value, it would be useful to know something about the variance of H . No such information is currently available and for this thesis a value of H will be considered significantly different if it differs by more than 0.05. This is based on experience applying R/S analysis to several sets of 1,000 generated random numbers.

Lyapunov Exponents

The **Lyapunov exponents** [10, Sec. 5.4] are a set of coefficients relating to chaotic dynamical systems which are much more prevalent in the literature than the Hurst coefficient. Briefly, these exponents measure the exponential rate of expansion or contraction of phase space in different directions as the system moves through time. A good, intuitive way

to think of these exponents is to envision the taffy pulling machine at a favorite amusement park. Place a drop of food coloring in the middle of the taffy (the phase space) and turn the machine on (the dynamics). The taffy is stretched in one direction and therefore is longer and thinner. The drop of coloring is now longer in the stretching direction (the first principal axis of the now three-dimensional ellipsoid) but shorter in the perpendicular directions (the other two principal axes). The number of Lyapunov exponents is equal to the number of dimensions of the phase space, in this case three. The exponential rate of stretching in this example corresponds to what is called the largest Lyapunov exponent. The degree of stretching, in this case negative stretching, corresponds to the remaining two Lyapunov exponents. Caution must be used to take the measurements during the pulling phase and not include the folding phase. The exponent measures stretching without folding by examining a very small drop of food coloring. In theory this is accomplished by choosing an arbitrarily small initial drop. In practice, no such drop can exist so an approximation must be made. The solution,

when folding is encountered, is to select a new set of points closer together which hold most of the information that the original points being iterated possessed. This is accomplished by adding a new, very small, drop of coloring which is essentially a scaled down version of the current (stretched and about to be folded) drop.

For those readers familiar with phase space analysis of differential equations, it may be useful to relate Lyapunov exponents to eigenvalues and eigenvectors. In phase-space analysis of second-order systems, fixed points are identified and eigenvalues and eigenvectors are used to build a "phase sketch" which gives qualitative information about the trajectory of a given starting point and the stability of the fixed points. The real part of the eigenvalues describes the rate at which a given trajectory moves toward or away from a fixed point. Whereas the eigenvalues describe behavior only near a fixed point, the Lyapunov exponent generalizes this idea to orbits which may not be fixed or periodic points. Formally, the Lyapunov exponents are defined by evolving an infinitesimal n -sphere in phase space. The

n -sphere then becomes an n -ellipsoid. From this ellipsoid, the i^{th} one-dimensional Lyapunov exponent is defined in terms of the length of the principal axis of the ellipsoid, $p_i(t)$ as:

$$\lambda_i = \lim_{t \rightarrow \infty} \frac{1}{t} \log_2 \frac{p_i(t)}{p_i(0)}$$

The difficulty in calculating the Lyapunov exponents using only a finite amount of data is that infinitesimal n -spheres can not be used. The object of using an infinitesimal n -sphere is to measure only the stretching and contracting of phase space without including any folding of the phase space that occurs. The method of numerically estimating the largest Lyapunov exponent used in this paper is the fixed evolution time method described in [11].

The procedure for estimating the largest exponent begins by constructing a phase portrait from the given time series using the method of delay coordinates. The point on the attractor in the constructed phase space that is closest (in Euclidean distance) to the initial point in the series is selected. This point is called the initial neighboring point. The distance

between these two points is labeled $L(t_0)$. After some evolution time, δt , this initial length has evolved to a new value $L'(t_1)$ with $t_1 = t_0 + \delta t$. The notation $L'(\cdot)$ distinguishes the distance between evolved points at a given time from the distance, $L(\cdot)$, between replacement points at the same time. The evolution time, δt , is chosen small enough that only the local stretching is measured without including any folding of phase space. Then, a replacement point for the evolved initial neighboring point is chosen which is close to the evolved initial point and whose angular separation from the evolved neighboring point is small. This new pair of points, the evolved initial point and the replacement point, approximates the outcome of evolving an initial pair of points whose separation was smaller than $L(t_0)$. This process continues until the initial point is evolved through the entire data set. In this way, the result approximates starting with a pair of points which are arbitrarily close. The largest Lyapunov exponent is then estimated by:

$$\lambda_1 = \frac{1}{t_m - t_0} \sum_{k=1}^M \log_2 \frac{L'(t_k)}{L(t_{k-1})}$$

In the limiting case of an infinite amount of noise free data, this method uses replacement vectors of infinitesimal length and with zero angular separation and so gives the exponent by definition. Some caution should be used when applying this method in order to minimize errors induced by the choice of replacement vectors.

The largest Lyapunov exponent is the most frequently used because it must be positive to have chaos in the system. When all exponents are negative it is an indication that the system, regardless of starting point, will be forced toward a stable solution. In the taffy example, this corresponds to negative stretching in all three directions. The result would be a large stable mass with no mixing or pulling or folding.

The analysis in this paper uses the largest Lyapunov exponent in two ways. First, the sign of the exponent must be positive for there to be chaos. Second, the magnitude of the largest exponent tells the rate at which information about the initial state of the system is lost through time. This is used to determine how far into the future the system can be

predicted given the measurement precision of the initial condition. For example, given two data sets whose measurements have the same precision, the set with the smaller exponent can be predicted further into the future.

Correlation Integral and Dimension

The final concept introduced is related to how many dimensions, or degrees of freedom, the system has. Degrees of freedom are basically the number of independent variables required to describe the system. A large number of degrees of freedom becomes essentially random, therefore a small number of dimensions are essential to be able to effectively model the system. The **correlation integral** [12] provides some help answering the question of how many dimensions.

The correlation integral is calculated from a given time series, $\{x_t\}$, by first constructing the phase portrait using the method of delay coordinates described previously [13]. The points on the reconstructed attractor are labeled z_1, z_2, \dots, z_n with $z_i = (x_{i+k}, x_{i+k-1}, \dots, x_i)$ where $k+1$ is the embedding dimension and n is equal to the number of points in

the time series minus k . From the constructed phase portrait, the following equation is used to calculate the correlation integral:

$$C(\varepsilon, n) = \frac{1}{n^2} \cdot \sum_{j=1}^n \sum_{i=1}^n U(\varepsilon - |z_i - z_j|)$$

In this equation, $U(\cdot)$ is the unit step function: $U(x) = 1$ if x is positive and 0 otherwise. The value of $C(\varepsilon, n)$ gives the probability that a randomly chosen pair of points on the attractor are separated by less than ε . Letting n tend to infinity, this equation defines $C(\varepsilon)$ according to:

$$C(\varepsilon) = \lim_{n \rightarrow \infty} C(\varepsilon, n)$$

The correlation integral leads to what is called the correlation dimension, D_2 , which is defined as:

$$D_2 = \lim_{\varepsilon \rightarrow 0} \frac{\ln C(\varepsilon)}{\ln \varepsilon}$$

For a finite amount of data, this dimension is approximated by the β_1 coefficient in the regression equation:

$$\ln(\varepsilon) = \beta_0 + \beta_1 \cdot \ln C(\varepsilon, n)$$

This equation holds for values of ε between $\min(|\mathbf{z}_i - \mathbf{z}_j|)$ and $\max(|\mathbf{z}_i - \mathbf{z}_j|)$. Further discussion of the correlation dimension can be found in [2].

The correlation dimension gives the minimum number of variables that will be required to model the system. Note that it does not give information about how many variables are enough, it only tells how many are not enough. For example, a correlation dimension of 2.1 indicates at least three variables required. The real system could require 250 variables. A correlation dimension of 250 means at least 250 variables are required and it may be time to look for a new data set.

Definitions of Chaos

This final section presents several authors' definitions of chaos. The definitions are not essential to the work in this thesis and are provided primarily to show the diversity in interpretations of

what is meant by "chaotic". The first definition comes from [3]. In their definition "Deterministic processes that look random are called deterministic chaos." More rigorously, the series $\{a_t\}$ has a deterministic chaotic explanation if there exists a system, (h, F, x_0) , such that h maps R^n to R , F maps R^n to R^n , $a_t = h(x_t)$, $x_{t+1} = F(x_t)$ and x_0 is the initial condition at $t=0$. The map F is deterministic, the state space is n -dimensional, all trajectories $\{x_t\}$ lie on an attractor A , and two nearby trajectories on A locally diverge. Local divergence is formalized by requiring the largest Lyapunov exponent to be positive.

A more topological definition is given [6]. The definition of chaos requires a few preliminary definitions.

Sensitive dependence on initial conditions

A function f that maps a set J to itself has sensitive dependence on initial conditions if there exists $\delta > 0$ such that, for any $x \in J$ and any neighborhood N of x , there exists $y \in N$ and $n \geq 0$ such that $|f^n(x) - f^n(y)| > \delta$.

Topological transitivity

A function f that maps a set J to itself is said to be topologically transitive if for any pair of open sets $U, V \subset J$ there exists $k > 0$ such that $f^k(U) \cap V \neq \emptyset$.

Periodic point

A point x is a periodic point of period n if $f^n(x) = x$.

Dense set

The subset, U , of periodic points in the attractor, S , is said to be dense in S if $\text{closure}(U) = S$.

With these definitions, a function f that maps a set V to itself is said to be chaotic if:

1. f has sensitive dependence on initial conditions.
2. f is topologically transitive.
3. periodic points are dense in V .

Devaney summarizes saying a chaotic map has unpredictability, indecomposability, and an element of regularity. Unpredictability and regularity may at

first seem incompatible. Basically, if the current state could be identified with infinite precision, the entire fate of the system could be determined. This is regularity. Since this is impossible and only finite accuracy is possible, the predictive power is limited. Thus, the system is ultimately unpredictable.

Harrel provides symptoms of chaos rather than a strict definition [4]. These symptoms are

- a) A positive Lyapunov exponent,
- b) The mixing property, and
- c) fractal geometry.

Ordinarily, he says, a chaotic dynamical system will exhibit at least two of these signs. The "mixing" property he uses states that a dynamical system is mixing if for any pair of measurable sets A and B ,

$$\lim_{t \rightarrow \infty} \mu(\phi_t^{-1}(A) \cap B) = \mu(A) \cdot \mu(B).$$

In this definition, μ is an invariant measure on the state space and $\phi_t(\cdot)$ is the flow of the dynamical system.

This completes the development of key ideas that are used in the next chapter which describes the

application of these tools to testing for randomness
in data and building a dynamical model of systems
which show evidence of a deterministic structure.

CHAPTER THREE

METHODS

Introduction

Chapter 2 provided an introduction to the world called Chaos. This chapter describes how to analyze a time series and then build and test a deterministic model of the series.

Overview

The method presented in this chapter evolved from an idea presented by Dr. William Lesso in a paper [14] which described modeling the sunspot numbers as a deterministic nonlinear system. This method consists of four stages: (1) graphical analysis, (2) application of diagnostic tools, (3) development of a deterministic model, and (4) evaluation of the effectiveness of the model in forecasting.

Graphical analysis

When presented with a new (one dimensional) series, two graphical analysis tools are used to gain a better understanding of the nature the data than is available by simply looking at a long list of numbers. The first plot examined is the time plot. First, the

series is plotted for the entire time domain. If the number of observations is large, perhaps more than 200, it may be appropriate to make several plots using smaller subdomains. This technique reveals more detail and provides more information about the structure of the data.

The other graphical tool applied is the phase portrait using a two-dimensional phase space and a lag period of one. Further variations on the phase portrait will be explored later in the analysis. The goal at this point is only to get a feel for the nature of the system being studied, not to analyze it.

Visual analysis of these simple graphs may yield ideas about the structure of the data. The next step is to use diagnostic tools to uncover firm evidence as to the true nature of the system.

Diagnostic Tools

The main objective in applying the following diagnostic tools is to determine whether the data being analyzed may be the result of a low-dimensional, deterministic, chaotic dynamical system. Throughout this thesis, this is referred to as the deterministic hypothesis. The other objective is to obtain

quantitative information about the data which will be useful in building a forecasting model.

In looking at the time plot, trends and patterns may or may not be evident in the data. Estimation of the Hurst coefficient provides much better information about the presence of trends than a simple visual analysis of the time plot. A Hurst coefficient which is not significantly different from the expected Hurst coefficient for a random process indicates that the range of the data scales at the same rate as for a random process and therefore it is likely that the data are not the result of a deterministic system. A coefficient value which is significantly different indicates then, that it is likely the system being studied is deterministic.

Note that a Hurst coefficient significantly different from the expected coefficient for a random system is neither necessary nor sufficient to conclude that the system is deterministic. Rather, it indicates that the scaling is typical of that for a deterministic system and therefore it is likely the system is deterministic. A Hurst coefficient which is significantly different from the expected coefficient

for a random system is considered evidence in support of the deterministic hypothesis. The BASIC code used to compute the Hurst coefficient is found in Appendix A.

The next diagnostic tool applied determines if the data are spatially correlated. Spatial correlation means that as the data are embedded in progressively higher dimensional phase spaces using the method of delay coordinates, the dimension of the reconstructed attractor stabilizes to a constant value. Strictly speaking, the dimension of the attractor used should be the box counting dimension. This dimension is difficult to calculate from a data set, however, and the correlation dimension is much less expensive to compute and provides a reasonable estimation of the box counting dimension. The Turbo Pascal code for estimating the correlation dimension is supplied in Appendix B.

Finding that the data is spatially correlated is considered support for the deterministic hypothesis. If the correlation dimension continues to increase with embedding dimension, then the reconstructed attractor essentially fills whatever space it is

embedded in. This is a common characteristic of random systems.

In addition to requiring the data to be spatially correlated, for this analysis the correlation dimension must converge to a small value. For this thesis, values less than five are considered small. Note that the dimension of the attractor can not be larger than the number of independent variables in the system which generated the data. For this reason, the minimum number of variables required to model the system is given by the smallest integer greater than the correlation dimension. The choice of five for a maximum correlation dimension is somewhat arbitrary and is chosen based on a belief that systems with more than five independent variables are too complex to attempt to model using the methods in this thesis. One final use of the correlation dimension is that non-integer values of this dimension indicate that the attractor may have a fractal structure which is a common symptom, although no guarantee of, a chaotic system.

The next diagnostic tool applied uses Micro TSP [15], a time series analysis package, to compute the

autocorrelations and partial correlations for the data set. Briefly, the i^{th} autocorrelation describes how much a given value of the series can be explained using the value of the series lagged i periods including the effects of all previous lag periods. The i^{th} partial correlation provides similar information but is calculating with the effects of all other lag variables held constant. The correlation dimension describes the spatial correlation while the autocorrelations describe the dynamic correlation of the data. A complete treatment of this subject is found in [1].

If a deterministic structure exists in the data, it should be seen as relatively strong values for the autocorrelation function. Strong values of the autocorrelations are therefore considered evidence in support of the deterministic hypothesis. One other benefit of the autocorrelation function is it may suggest a more appropriate lag period for which to generate a phase portrait. If, for example, the autocorrelation function shows strong correlation with the variable lagged three periods and little correlation with the first two lags, it is appropriate

to generate a phase portrait using a lag period of three. Finally, the correlation dimension gives information about how many variables to include in a forecasting model; the lag periods with the strongest autocorrelations are considered the best choices to include in the model.

The tests described to this point have been attempts to find support for the deterministic hypothesis. The next diagnostic tool, estimation of the largest Lyapunov exponent tests not for structure but for the presence of chaos. As described earlier, the largest Lyapunov exponent must be positive for the system to have sensitive dependence on initial conditions. If the largest Lyapunov exponent is found to be negative, then the data is not chaotic and the method described here is not appropriate for modeling that data. Therefore, a forecasting model is built only for systems which have a positive largest Lyapunov exponent. The BASIC computer code for estimating the largest Lyapunov exponent is given in Appendix C.

The final diagnostic test applied extends what Peters calls the Shuffle Diagnostic to include not

only the Hurst coefficient but also the autocorrelations.

Peters recommends shuffling the data set and recomputing the Hurst coefficient (the Shuffling diagnostic) to determine whether the original Hurst coefficient is due to some statistical phenomena of the data [16, p.75]. If the Hurst coefficient for the shuffled data is significantly closer to the expected value of H for a random process, the apparent structure may have been destroyed by the shuffling process and this supports the deterministic hypothesis. If the coefficient does not change after shuffling, the data should be examined to attempt to determine another explanation for the anomalous value obtained.

This same idea is then applied using the autocorrelations. If the autocorrelations observed for the original data set are no longer present in the shuffled data, then the original autocorrelations are most likely due to a deterministic structure and not a statistical phenomena. A significant reduction in the autocorrelations after shuffling is then considered support for the deterministic hypothesis.

For the analysis in this paper, the shuffling is accomplished by first importing the data into Microsoft Excel 4.0, a spreadsheet software package. Then, a sequence of random numbers is generated and listed with the time series as pairs. These pairs are then sorted in ascending order according to the value of the random number. To reduce the possibility of artificial structure being induced by shuffling according to pseudo-random numbers, the shuffling is accomplished twenty times.

With the above tests completed, the analysis now becomes somewhat more subjective and qualitative. All of the results are considered and a determination is made whether the results support the hypothesis that the data are the result of a low-dimensional deterministic system. In summary, results which support the deterministic hypothesis include: (1) a Hurst coefficient significantly different from the value for a random process which moves close to this value after shuffling, (2) a low (less than five) correlation dimension, (3) strong autocorrelations which are then no longer present after shuffling, and (4) a positive largest Lyapunov exponent. If there is

support for the deterministic hypothesis, a deterministic model is built using the method described in the following section.

Modeling the Data

The correlation dimension gives the first input to the form of the model to be tried. As mentioned earlier, the model must have a number of variables no less than the correlation dimension. The variables considered will be lags of the time series. That is, the model is of the form $x_t = f(x_{t-a_1}, x_{t-a_2}, \dots, x_{t-a_n})$ with $0 < a_1 < a_2 \dots < a_n$ and n is greater than or equal to the correlation dimension. Here the notation x_{t-a_i} is used to represent the lags, that is the value of the time series lagged a_i periods. The lag periods with the strongest autocorrelations are the first choice of variables to include in this model.

To choose an initial regression model for the selected variables, phase portraits for the lag periods used are generated and visually analyzed in hopes of finding a recognizable pattern. Clearly, higher dimensional systems pose greater difficulty for this graphical analysis.

The best standard model, such as a polynomial, trigonometric function, exponential, conic section or a combination (additive or multiplicative) of these, suggested by the graphical analysis is tried for the first regression model. For example, if the phase portrait for the first lag variable looks like a parabola, then the first regression model tried is:

$$x_t = \beta_0 + \beta_1 \cdot x_{t-1} + \beta_2 \cdot x_{t-1}^2$$

This equation is then applied in a linear regression using MicroTSP [15] to estimate the regression coefficients. This equation, with the values of the regression coefficients inserted, then is the forecasting model.

As a more sophisticated example, suppose the phase portrait has the basic shape of an ellipse. The regression equation in this case is derived from the general equation for an ellipse:

$$A \cdot x^2 + B \cdot x \cdot y + C \cdot y^2 + D \cdot x + E \cdot y + F = 0$$

The regression equation is derived by first substituting x_t for y and x_{t-1} for x . Then the y^2 term

is moved to the right side of the equation. Finally, since the equation is not changed by dividing by a non-zero constant, the equation is divided by $-C$ and, reordering terms, the regression model is given by:

$$x_t^2 = \beta_0 + \beta_1 \cdot x_t + \beta_2 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-1} + \beta_4 \cdot x_t \cdot x_{t-1}$$

As in the previous example, the least squares regression returns the estimated coefficients. In this case, the quadratic formula must be applied to solve explicitly for x_t as a function of x_{t-1} when building the forecasting model. The forecasting model for this system is:

$$x_t = \frac{1}{2 \cdot a} \cdot -b \pm (b^2 - 4 \cdot a \cdot c)^{0.5}$$

$$a = 1.0$$

$$b = -(\beta_1 + \beta_4 \cdot x_{t-1})$$

$$c = -(\beta_0 + \beta_2 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-1})$$

When to use the positive and negative values of the square root in the forecasting example depends on the system and this issue is discussed by way of example

in Chapter 4 during the analysis of the Rossler equations.

When the least squares regression is done, a series of residuals is generated. These residuals are the difference between the actual values and the values estimated by the regression. These residuals are tested for autocorrelations to look for ways to improve the original regression equation. If no autocorrelations are present, the residuals are essentially random values and it is likely no improvements can be made to the original model. If autocorrelations are present, the next step is to generate a phase portrait for the residual series using the best lag periods suggested by the autocorrelations. This information is then used in the same way the original model was built to add terms to the original regression equation. As an example of this, consider again the earlier example of a phase portrait with the shape of a parabola. If the phase plot of the residual series using a lag period of two is close to a straight line, the new regression model becomes:

$$x_t = \beta_0 + \beta_1 \cdot x_{t-1} + \beta_2 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-2}$$

This process continues until at least one of the following conditions exist: (1) the residuals are no longer autocorrelated, (2) the fit is considered adequate, or (3) the residuals give no suggestions for improvement of the model. For the second case, the fit is considered adequate if the R-squared value for the regression is close to 1.0. The value, R-squared, is a measure of the goodness of fit of the regression. If the R-squared value is 1.0, the regression fits the data perfectly. An R-squared value of zero indicates the regression fits the data no better than a constant function equal to the sample mean [15]. In this thesis, an R-squared value is considered close to 1.0 if it is greater than 0.99.

Testing the Model

The goal of this research is to show that, for some time series, it is more effective to forecast the data using a deterministic chaotic model than using standard time series techniques. The standard time series model used for comparison is a Box-Jenkins ARIMA model. Only the results of building the ARIMA model are included here and for discussion of this topic the reader is referred again to [1].

To accomplish this test of effectiveness, the chaotic deterministic model is used to forecast the series. The starting point for the forecast is the end of the data set used in building the model. Since the generating functions for the data are known, the data set can be extended and the forecast values are then compared to the extended actual values.

The data set is also modeled using an ARIMA model and this model is used to forecast the data starting at the same point as for the chaotic deterministic model forecast. To compare the predictive power of the two models, the number of iterations before the predicted value differs from the actual by more than a subjectively determined "significant" amount is counted. The deterministic model is considered more effective in forecasting if the number of iterations before "divergence" is smaller than that for the ARIMA model. In this thesis, the subjectively determined "significant" amount depends on the system being studied but is on the order of ten percent of the maximum range of the series.

Summary

This chapter described how to apply the tools presented in Chapter 2 as well as others introduced in this chapter to build and test a deterministic model to forecast a data set which is suspected to be chaotic. In the next chapter, this method is applied to time series generated from three known chaotic functions.

CHAPTER FOUR

APPLICATION

Introduction

This chapter contains the details of applying the previously described analysis to sets of data generated from three known chaotic functions. The functions used to generate this data are the logistic equation, the Henon equations and the Rossler equations.

Logistic Function

The first sample data sets analyzed were generated from the logistic equation, that is $x_{t+1} = \lambda \cdot x_t \cdot (1 - x_t)$. This function is analyzed first because it is the simplest of those studied in this chapter. The system is be studied for two values of the parameter λ , $\lambda = 3.8$ and $\lambda = 3.92$. These are both in the chaotic regime and the two sets are chosen to demonstrate that, while the behavior of the system is extremely sensitive to parameter changes, the analysis method is robust. One thousand data points are used since this number of points lessens the risk of studying some transient behavior pattern that is not

typical while keeping reasonable the computation time for the data intensive tests that follow. Each step of the analysis is done for both parameter values before proceeding to the next step. The decision to present the two this way rather than as two independent analyses is intended to facilitate comparison.

The first step in this analysis is to generate a time plot and a phase portrait in two dimensions using a lag of one period. These plots deserve at least a cursory inspection, looking for any obvious trends, patterns and cycles. The time plots for the two parameter values shown in Figure 4.1 and Figure 4.2 show some signs of patterns but yield no obvious evidence of structure.

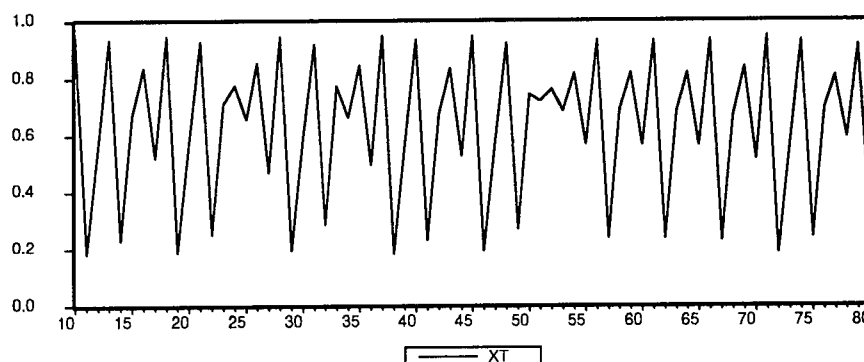


Figure 4.1 Time plot for $\lambda = 3.8$

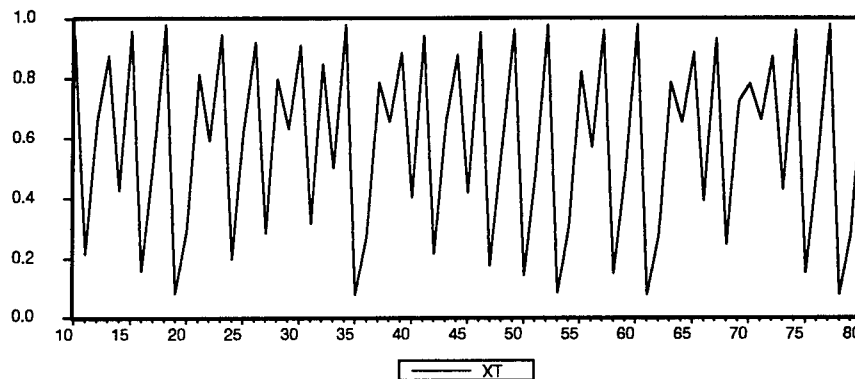


Figure 4.2 Time plot for $\lambda = 3.92$

A comparison of the two time plots shows some similarity but the behavior of the system is clearly sensitive to parameter changes.

The phase portraits given in Figure 4.3 and Figure 4.4 definitely show strong evidence of structure. Additionally, while the time plots showed only a hint of similarity, the phase portraits appear nearly identical. These plots are key tools for building the dynamical models and their usefulness is demonstrated later in the analysis when the models are built and tested.

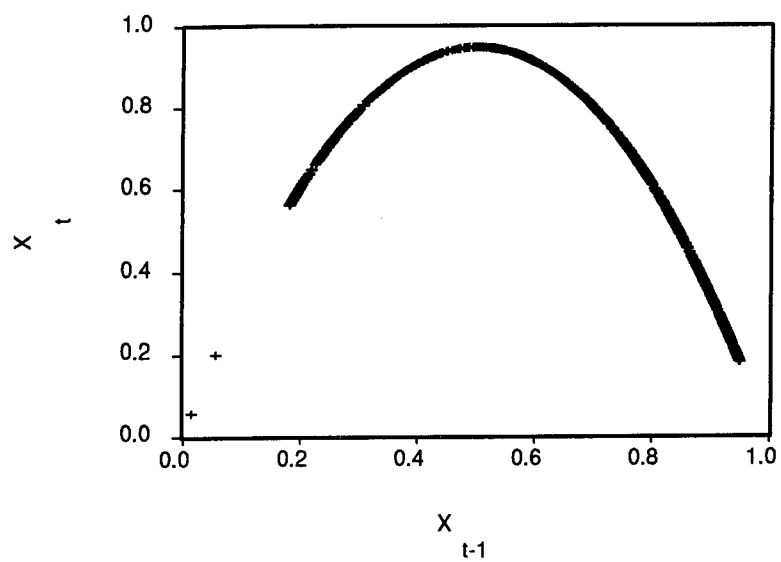


Figure 4.3 Phase Portrait for $\lambda = 3.8$

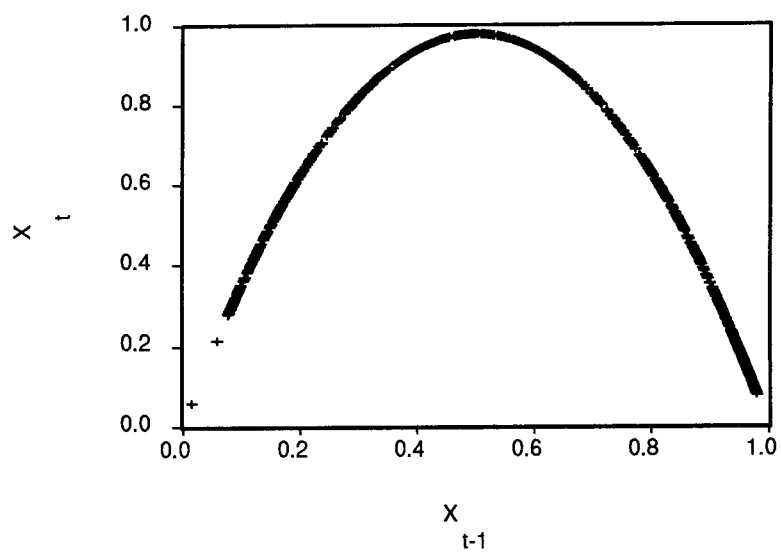


Figure 4.4 Phase portrait for $\lambda = 3.92$

The results of calculating the Hurst coefficient for both the original and shuffled data sets as well as the correlation dimension and largest Lyapunov exponent are summarized in Table 4.1.

Hurst coefficient	0.437	0.441
Hurst coefficient for scrambled data set	0.492	0.514
Correlation dimension	0.89	0.94
Largest Lyapunov exponent	0.311	0.271

Table 4.1 Summary of Diagnostic tools

All of these results support the deterministic hypothesis. Comparing, both show that shuffling appears to have removed the antipersistence and both may have a low-dimensional structure. The Hurst coefficients show that for $\lambda = 3.92$, the time series is "more random" and the Lyapunov exponents show that the series for $\lambda = 3.8$ is "more chaotic".

The time series' are now tested for auto-correlations and partial correlations using MicroTSP [15]. The results are given in Table 4.2 and Table 4.3. Both show strong dependence on the lag-one variable and some dependence on the lag-two variable.



Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 -0.596	-0.596	355.85	0.000
		2 0.195	-0.248	394.15	0.000
		3 0.009	0.011	394.23	0.000
		4 -0.121	-0.100	408.86	0.000
		5 0.136	0.003	427.42	0.000
		6 -0.148	-0.097	449.58	0.000
		7 0.071	-0.089	454.71	0.000
		8 0.017	0.006	455.01	0.000
		9 -0.010	0.067	455.10	0.000
		10 0.021	0.046	455.55	0.000
		11 -0.023	0.012	456.10	0.000
		12 0.039	0.042	457.64	0.000

Table 4.2 Autocorrelations for $\lambda = 3.8$

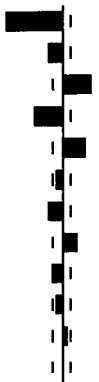

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 -0.423	-0.423	179.14	0.000
		2 -0.102	-0.342	189.59	0.000
		3 0.224	0.030	240.17	0.000
		4 -0.211	-0.147	285.13	0.000
		5 0.180	0.112	317.71	0.000
		6 -0.042	0.025	319.48	0.000
		7 -0.093	-0.017	328.28	0.000
		8 0.113	0.002	341.15	0.000
		9 -0.061	-0.001	344.93	0.000
		10 -0.041	-0.074	346.66	0.000
		11 0.033	-0.068	347.77	0.000
		12 0.015	0.016	347.99	0.000

Table 4.3 Autocorrelations for $\lambda = 3.92$

Both are evidence to support the deterministic hypothesis. Additionally, no other lags show a

stronger dependence than is found on the first lag and this indicates that the first lag period is the most appropriate choice for building phase portraits and also dynamical models.

The autocorrelations and partial correlations for the shuffled data sets shown in Table 4.4 and Table 4.5 indicate that the shuffling process appears to have removed the structure in both cases and these results also support the deterministic hypothesis.

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 0.010	0.010	0.1059	0.745
		2 -0.014	-0.014	0.3092	0.857
		3 0.008	0.008	0.3745	0.945
		4 0.050	0.049	2.8616	0.581
		5 -0.071	-0.072	7.8924	0.162
		6 -0.002	0.001	7.8973	0.246
		7 -0.005	-0.007	7.9201	0.340
		8 -0.045	-0.046	9.9207	0.271
		9 -0.008	0.000	9.9855	0.352
		10 -0.001	-0.007	9.9870	0.442
		11 -0.001	-0.000	9.9892	0.531
		12 0.005	0.009	10.013	0.615

Table 4.4 Autocorrelations for shuffled data
for $\lambda = 3.8$

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
1		0.004	0.004	0.0138	0.906
2		0.014	0.014	0.2069	0.902
3		-0.045	-0.045	2.2029	0.531
4		-0.030	-0.030	3.1016	0.541
5		-0.024	-0.023	3.6801	0.596
6		-0.001	-0.002	3.6807	0.720
7		-0.011	-0.013	3.8061	0.802
8		-0.011	-0.014	3.9397	0.863
9		0.016	0.015	4.1878	0.899
10		-0.018	-0.020	4.5168	0.921
11		-0.033	-0.036	5.6445	0.896
12		0.061	0.062	9.3865	0.670

Table 4.5 Autocorrelations for shuffled data
for $\lambda = 3.92$

The Hurst coefficient for the parameter value $\lambda = 3.8$ is estimated to be 0.437. This indicates that the system is antipersistent. The web diagram (Figure 4.5) supports this conclusion as well. The antipersistence can be seen by noting that upward movements are more likely to be followed by downward movements and vice versa. For the parameter value $\lambda = 3.92$ the estimated Hurst coefficient is 0.441, again indicating antipersistence but to a lesser degree. This also can be seen in the web diagram (Figure 4.6) since upward movements are still more frequently followed by downward movements and vice

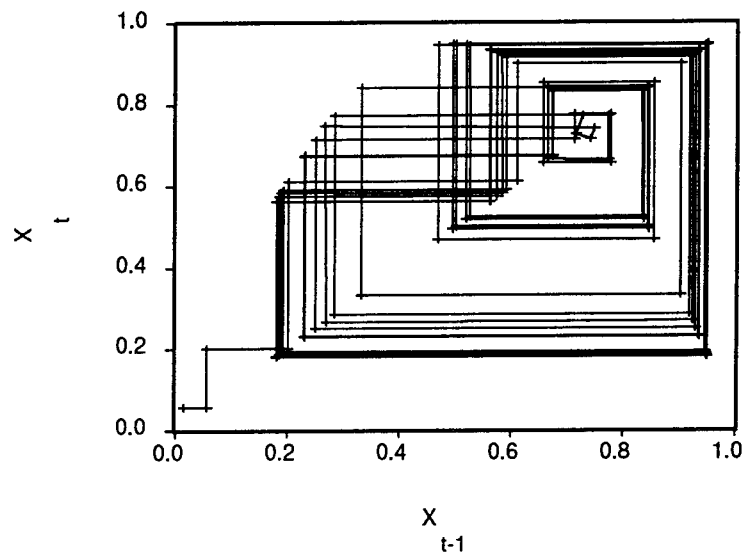


Figure 4.5 Web diagram for $\lambda = 3.8$

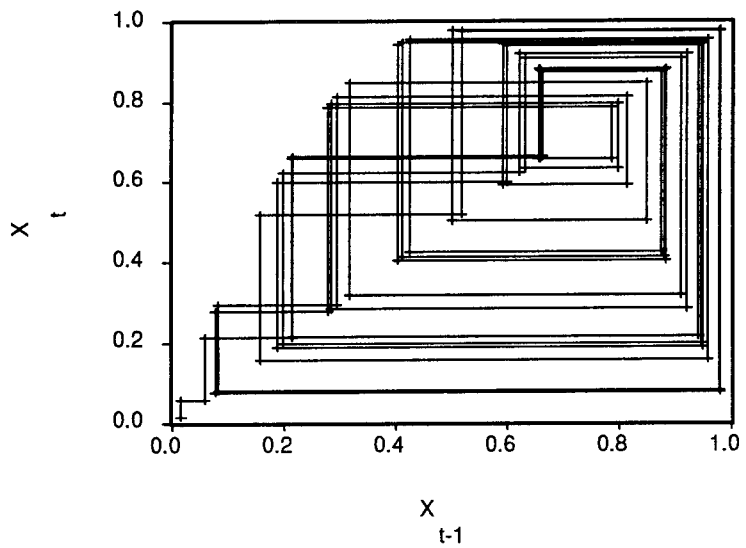


Figure 4.6 Web diagram for $\lambda = 3.92$

versa, but the ratio of trend reversing to trend reinforcing movements seems to be smaller.

The correlation dimensions show it may be possible to model both systems using only one variable and both the phase plots and autocorrelations show no evidence of a better lag variable than a lag of one period. Therefore, supported by this evidence, the first model for both systems is the one suggested by the phase plots, a lag-one quadratic:

$$x_t = \beta_0 + \beta_1 \cdot x_{t-1} + \beta_2 \cdot x_{t-1}^2$$

The results of the regression using Micro TSP [15] are shown for $\lambda = 3.8$ in Table 4.6 and for $\lambda = 3.92$ in Table 4.7. Note that both models fit their respective data sets with R -squared of 1.00 and the original coefficients (3.8 and 3.92) are exactly recovered. The data sets used in this analysis are rounded to six significant figures and therefore the constant terms in both regressions are smaller than the possible rounding error. For this reason, the dynamical models are built using a β_0 coefficient of zero.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
$(X_{t-1})^2$	-3.800000	4.14E-07	-9178322.	0.0000
X_{t-1}	3.800000	4.87E-07	7805314.	0.0000
C	-1.11E-07	1.26E-07	-0.880403	0.3789
R-squared	1.000000	Mean dependent var	0.639909	
Adjusted R-squared	1.000000	S.D. dependent var	0.249087	
S.E. of regression	6.87E-07	Akaike info criterion	-28.37796	
Sum squared resid	4.71E-10	Schwartz criterion	-28.36323	
Log likelihood	12760.27	F-statistic	6.55E+13	
Durbin-Watson stat	1.619505	Prob(F-statistic)	0.000000	

Table 4.6 Regression output for $\lambda = 3.8$

Variable	Coefficient	Std. Error	T-Statistic	Prob.
$(X_{t-1})^2$	-3.920000	6.13E-09	-6.39E+08	0.0000
X_{t-1}	3.920000	6.93E-09	5.65E+08	0.0000
C	7.26E-10	1.60E-09	0.454303	0.6497
R-squared	1.000000	Mean dependent var	0.583984	
Adjusted R-squared	1.000000	S.D. dependent var	0.306489	
S.E. of regression	1.38E-08	Akaike info criterion	-36.20071	
Sum squared resid	1.89E-13	Schwartz criterion	-36.18599	
Log likelihood	16684.42	F-statistic	2.48E+17	
Durbin-Watson stat	1.963275	Prob(F-statistic)	0.000000	

Table 4.7 Regression output for $\lambda = 3.92$

Since the generating functions are completely recovered for both cases, the models are not used to forecast the series and count iterations until divergence. The only necessary step to show the

effectiveness of the dynamical models is to demonstrate that the same effectiveness can not be achieved using ARIMA models.

The result of using MicroTSP [15] to build the ARIMA models are shown below in Table 4.8 and Table 4.9.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
C	0.640894	0.001981	323.5075	0.0000
AR(1)	0.054882	0.173685	0.315986	0.7521
AR(2)	0.177544	0.058182	3.051560	0.0023
MA(1)	-0.885156	0.175200	-5.052256	0.0000
MA(2)	0.141811	0.104945	1.351285	0.1769
R-squared	0.440328	Mean dependent var	0.640494	
Adjusted R-squared	0.438074	S.D. dependent var	0.248524	
S.E. of regression	0.186298	Akaike info criterion	-3.355819	
Sum squared resid	34.46398	Schwartz criterion	-3.331241	
Log likelihood	263.4528	F-statistic	195.3137	
Durbin-Watson stat	2.010500	Prob(F-statistic)	0.000000	

Table 4.8 ARIMA model regression for $\lambda = 3.8$

Variable	Coefficient	Std. Error	T-Statistic	Prob.
C	0.584668	0.004078	143.3781	0.0000
AR(1)	-0.898384	0.067144	-13.38000	0.0000
AR(2)	-0.330551	0.052807	-6.259585	0.0000
MA(1)	0.348409	0.069099	5.042166	0.0000
MA(2)	-0.233164	0.062674	-3.720246	0.0002
R-squared	0.294319	Mean dependent var	0.584882	
Adjusted R-squared	0.291476	S.D. dependent var	0.306118	
S.E. of regression	0.257671	Akaike info criterion	-2.707145	
Sum squared resid	65.92967	Schwartz criterion	-2.682567	
Log likelihood	-60.23543	F-statistic	103.5378	
Durbin-Watson stat	1.998270	Prob(F-statistic)	0.000000	

Table 4.9 ARIMA model regression for $\lambda = 3.92$

The plots given in Figure 4.7 and Figure 4.8 show the ARIMA forecast values and the actual extended series values for the two parameter values.

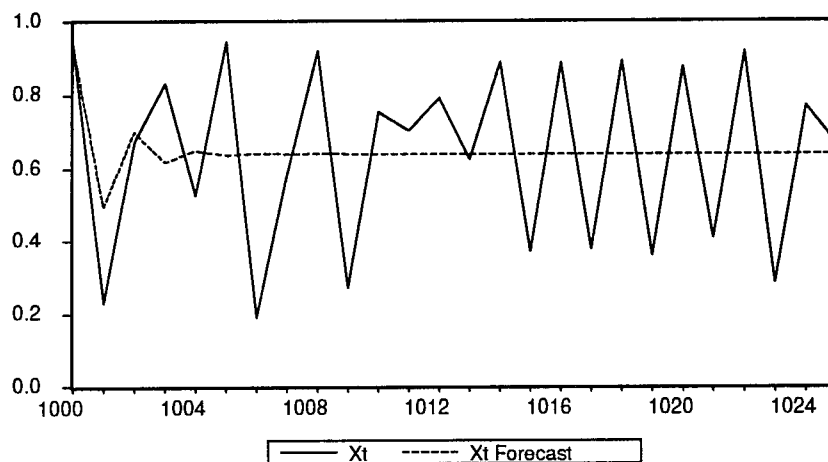


Figure 4.7 ARIMA forecast and actual values for $\lambda = 3.8$

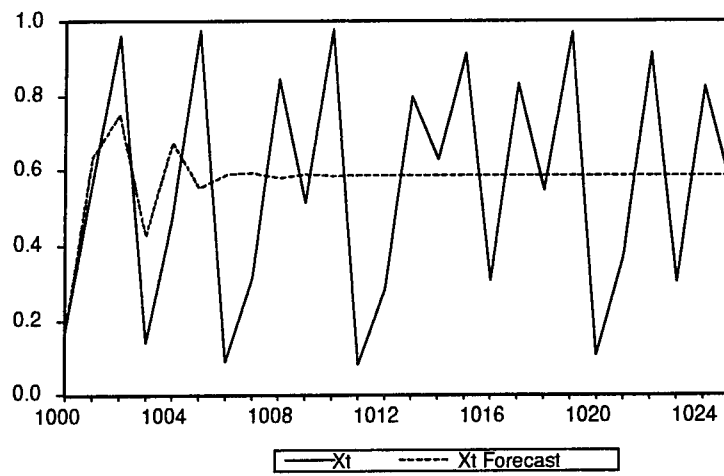


Figure 4.8 ARIMA forecast vs. actual values
for $\lambda = 3.92$

The difference between these two values is plotted in Figure 4.9 for $\lambda = 3.8$ and in Figure 4.10 for $\lambda = 3.92$.

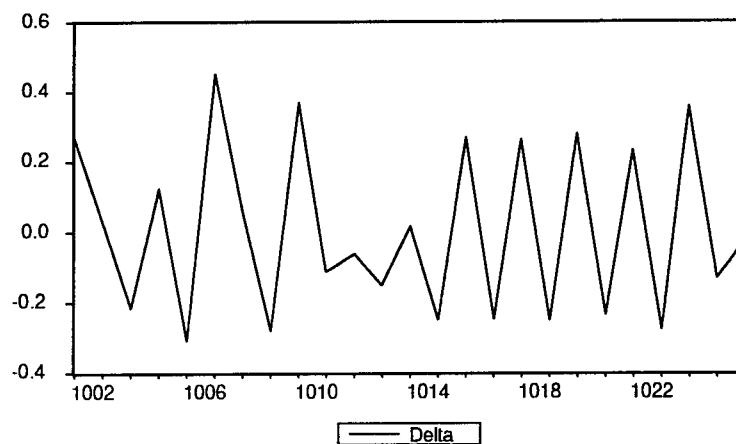


Figure 4.9 Plot of divergence for $\lambda = 3.8$

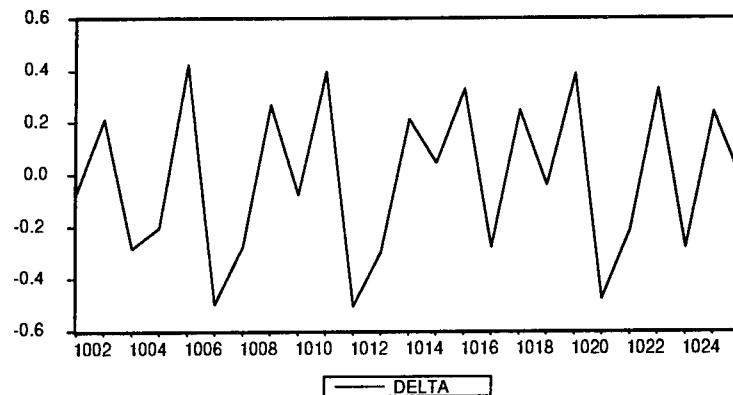


Figure 4.10 Plot of Divergence for $\lambda = 3.92$

From this result, it is clear that this function resists forecasting with ARIMA models. The difference between the forecast value and the actual value is larger than twenty percent of the range of the series for even the first iteration in the case of $\lambda = 3.8$. For the case of $\lambda = 3.92$ the forecast is better but still differs by more than ten percent of the range after only two iterations. This percentage of the maximum value is in some sense a percentage error. These results show that the function is significantly more effectively forecast using deterministic chaotic models.

Henon Equations

Data generated from the Henon equations are examined next in the analysis. The equations used to generate the data set are:

$$x_{t+1} = 1 - a \cdot x_t^2 + y_t$$

$$y_{t+1} = b \cdot x_t$$

The classic parameters, $a = 1.4$ and $b = 0.3$, are used with the initial point (0.5,0.5). The time series of x -values are analyzed. Note that the y -values are simply the x -values scaled by a constant value. Therefore, demonstrating the procedure for the x -values also demonstrates the procedure for the y -values.

The time plot (Figure 4.11) shows no readily apparent structure.

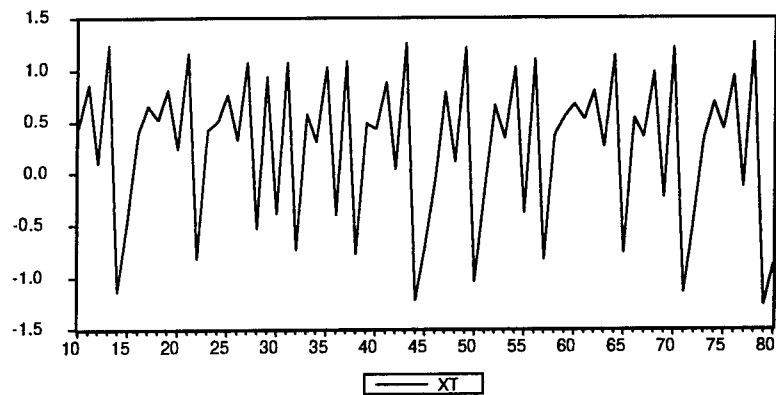


Figure 4.11 Time plot

The structure in the data becomes much more apparent in the phase portrait (Figure 4.12).

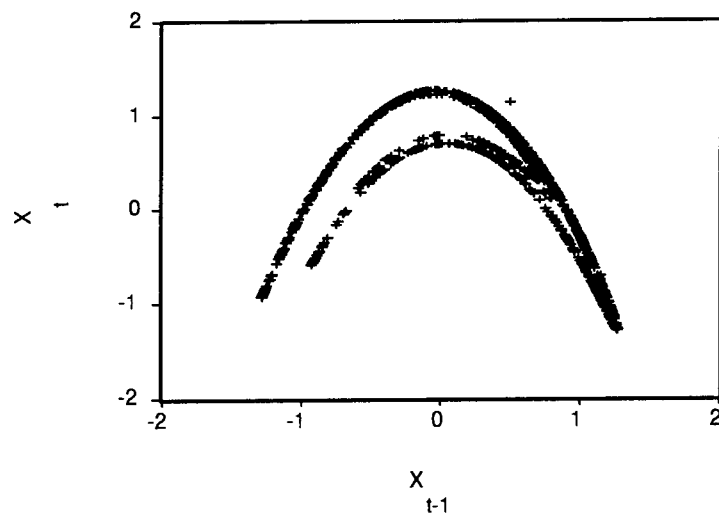


Figure 4.12 Phase portrait

This plot shows strong evidence of structure although the structure seems more complex than that previously observed for the logistic equation.

The result of computing the Hurst coefficient, the correlation dimension and the largest Lyapunov exponent are summarized in Table 4.10.

Hurst coefficient	0.435
Hurst coefficient for scrambled data set	0.622
Correlation dimension	1.23
Largest Lyapunov exponent	0.432

Table 4.10 Summary of Diagnostic tools

These results all support the deterministic hypothesis. The most questionable result is the Hurst coefficient for the shuffled data. As described earlier, the expected H for a random process with 1,000 data points is 0.558. Shuffling the data moves H closer to this value but not within the ± 0.06 window used to determine significance. The data set, when shuffled an additional twenty times, has a Hurst coefficient of 0.506 which is closer to the value anticipated. With this result, the shuffling

diagnostic for H seems to more strongly support the deterministic hypothesis.

Probably the most significant result here is the correlation dimension. While the correlation dimension for both parameter values of the logistic equation showed it may be possible to model the system with only one independent variable, the results shown above indicate that the model for this data set must include at least two independent variables.

A relatively strong dependence on the lag one and three periods is observed in the calculation of autocorrelations (Table 4.11). Additionally, there is a weaker dependence on lags two and four through six.

























Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 -0.302	-0.302	91.610	0.000
		2 0.252	0.177	155.47	0.000
		3 -0.359	-0.276	284.84	0.000
		4 0.008	-0.215	284.90	0.000
		5 -0.161	-0.124	310.84	0.000
		6 -0.045	-0.229	312.87	0.000
		7 0.149	0.056	335.21	0.000
		8 -0.016	-0.032	335.45	0.000
		9 0.151	-0.006	358.65	0.000
		10 -0.074	0.012	364.26	0.000
		11 0.029	-0.035	365.09	0.000
		12 -0.065	0.023	369.39	0.000

Table 4.11 Autocorrelations

This is evidence to support the deterministic hypothesis and also supports the choice of the lag one variable for the initial phase portrait. This choice of the lag one variable is based on the partial autocorrelations since these essentially treat each lag variable separated from the effects of the others. This is why the lag one variable, and not the lag three variable which has a higher autocorrelation, is chosen.

The autocorrelations are no longer present in the shuffled data set as is shown in Table 4.12. This result also supports the deterministic hypothesis.

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 0.024	0.024	0.5728	0.449
		2 0.030	0.030	1.4919	0.474
		3 0.028	0.027	2.2779	0.517
		4 0.027	0.025	3.0039	0.557
		5 -0.022	-0.025	3.4910	0.625
		6 0.038	0.037	4.9240	0.554
		7 0.015	0.013	5.1535	0.641
		8 0.008	0.006	5.2185	0.734
		9 -0.014	-0.016	5.4285	0.795
		10 0.022	0.019	5.9292	0.821
		11 0.038	0.039	7.3704	0.768
		12 -0.036	-0.039	8.6517	0.732

Table 4.12 Autocorrelations for shuffled data

The phase portrait for the data using a lag period of one suggests the same quadratic model used for the logistic equation as a starting point:

$$x_t = \beta_0 + \beta_1 \cdot x_{t-1} + \beta_2 \cdot x_{t-1}^2$$

There appears to be more than a simple lag-one quadratic in the phase portrait and the correlation dimension suggests at least one more independent variable is required to model the data. The hope is that the quadratic model is "close" and the residuals, the difference between actual and fitted values, from the regression will uncover the additional factors.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
$(x_{t-1})^2$	-1.348170	0.013349	-100.9920	0.0000
x_{t-1}	-0.098457	0.009237	-10.65911	0.0000
C	1.077341	0.009947	108.3128	0.0000
R-squared	0.919170	Mean dependent var	0.269063	
Adjusted R-squared	0.919008	S.D. dependent var	0.713320	
S.E. of regression	0.203004	Akaike info criterion	-3.186057	
Sum squared resid	41.04592	Schwartz criterion	-3.171322	
Log likelihood	176.9161	F-statistic	5663.102	
Durbin-Watson stat	1.881319	Prob(F-statistic)	0.000000	

Table 4.13 Regression output

The fit in Table 4.13 is good but the R -squared value is not close to 1.0, therefore the next step is to calculate the autocorrelations for the residual series to see if additional structure exists. The autocorrelations are shown in Table 4.14



Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 0.057	0.057	3.2026	0.074
		2 0.167	0.164	31.208	0.000
		3 -0.379	-0.408	175.15	0.000
		4 -0.165	-0.160	202.59	0.000
		5 -0.227	-0.082	254.59	0.000
		6 -0.057	-0.165	257.86	0.000
		7 0.146	0.112	279.35	0.000
		8 0.085	-0.045	286.64	0.000
		9 0.165	0.000	314.15	0.000
		10 -0.025	0.013	314.80	0.000
		11 -0.000	-0.024	314.80	0.000
		12 -0.065	0.055	319.04	0.000

Table 4.14 Autocorrelations for residual series

The strongest autocorrelation is in the third lag period. A plot of the residuals against x_t lagged three periods (Figure 4.13) shows similar complexity to the original phase portrait and suggests looking at other lag periods. The next strongest autocorrelation

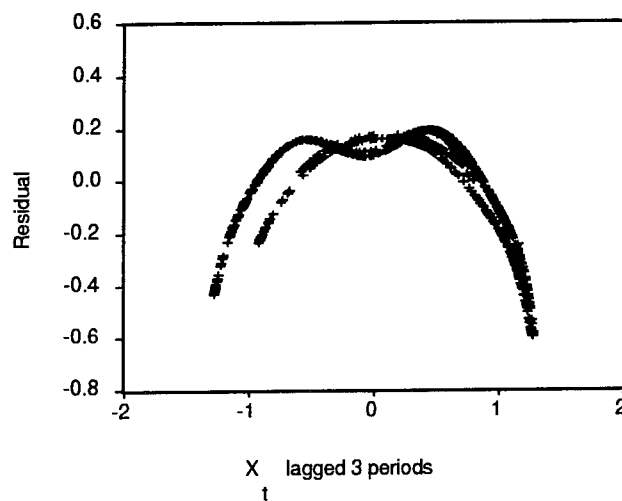


Figure 4.13 Residual vs. lag three variable

is in the second lag period. The plot of the residuals against x_t lagged two period is given in Figure 4.14.

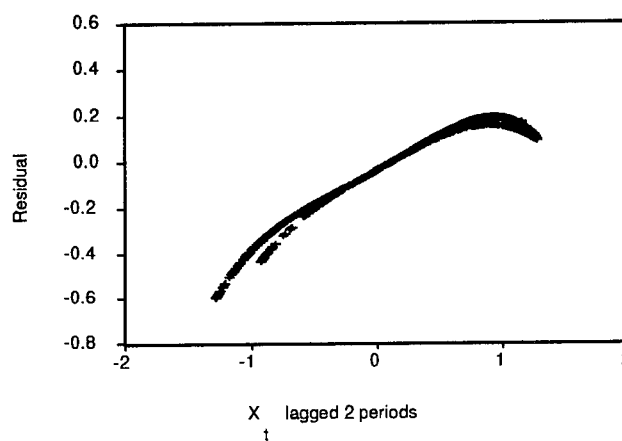


Figure 4.14 Residual vs. lag two variable

This shows strong evidence of the presence of a linear lag two term.

Based on the information in Figure 4.14 and the fact that the correlation dimension indicates the need for at least two independent variables, the original lag one quadratic model is modified to include the linear term in the second lag, that is the second regression model is:

$$x_t = \beta_0 + \beta_1 \cdot x_{t-1} + \beta_2 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-2}$$

The result of the regression to estimate the coefficients in the new model is shown in Table 4.15.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
$(x_{t-1})^2$	-1.400000	2.50E-07	-5601417.	0.0000
x_{t-1}	4.39E-07	1.81E-07	2.424900	0.0155
x_{t-2}	0.300000	1.77E-07	1694852.	0.0000
C	1.000000	1.90E-07	5255166.	0.0000
R-squared	1.000000	Mean dependent var	0.268180	
Adjusted R-squared	1.000000	S.D. dependent var	0.713131	
S.E. of regression	3.77E-06	Akaike info criterion	-24.97267	
Sum squared resid	1.41E-08	Schwartz criterion	-24.95301	
Log likelihood	11049.26	F-statistic	1.19E+13	
Durbin-Watson stat	2.416986	Prob(F-statistic)	0.000000	

Table 4.15 Regression output for second model

This result has an *R*-squared value of 1.0 and therefore no further improvements are attempted. The coefficient of the linear term in lag-one is smaller than the possible rounding error for the data set which is rounded to six significant figures and therefore is considered to be zero.

Note that if the formula for the *y*-values in the generating equation is substituted into the formula for the *x*-values the result has the form of the second regression model. That is, the system of equations:

$$x_{t+1} = 1 - a \cdot x_t^2 + y_t$$

$$y_{t+1} = b \cdot x_t$$

can be rewritten as:

$$x_{t+1} = 1 - a \cdot x_t^2 + y_t$$

$$y_t = b \cdot x_{t-1}$$

And then with substitution becomes:

$$x_{t+1} = 1 - a \cdot x_t^2 + b \cdot x_{t-1}$$

Using this representation of the system of equations shows that coefficients resulting from the regression are exactly those used to generate the data.

As with the analysis of the logistic equation, since the generating equation is completely recovered, the only step remaining to verify the effectiveness of this method is to show the same result does not occur using the ARIMA model.

The regression result of building the ARIMA model is given in Table 4.16.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
C	0.268556	0.008350	32.16327	0.0000
AR(1)	0.559630	0.064985	8.611734	0.0000
AR(2)	0.127423	0.054981	2.317575	0.0207
AR(3)	-0.443136	0.031296	-14.15928	0.0000
MA(1)	-0.885995	0.070384	-12.58799	0.0000
MA(2)	0.209848	0.069589	3.015527	0.0026
R-squared	0.255736	Mean dependent var	0.269153	
Adjusted R-squared	0.251980	S.D. dependent var	0.712827	
S.E. of regression	0.616510	Akaike info criterion	-0.961360	
Sum squared resid	376.6643	Schwartz criterion	-0.931843	
Log likelihood	-929.4435	F-statistic	68.10321	
Durbin-Watson stat	1.985491	Prob(F-statistic)	0.000000	

Table 4.16 Regression result for ARIMA model

This model is then used to forecast the data and the plot of the actual and forecast values is given in Figure 4.15 below. The divergence is calculated and this series is plotted in Figure 4.16. The range for the original series is approximately 3.0 and, using

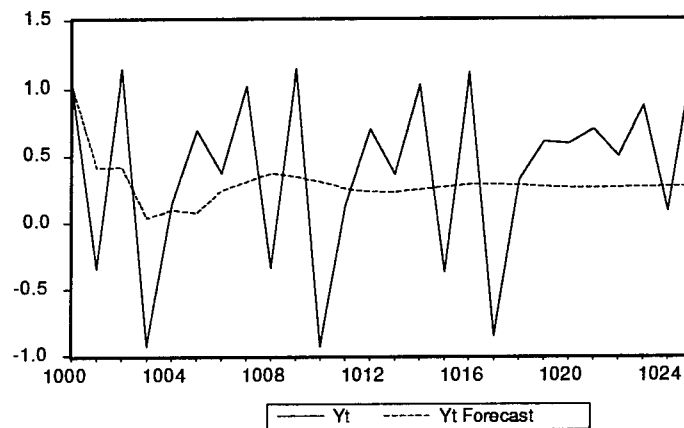


Figure 4.15 Forecast and actual values

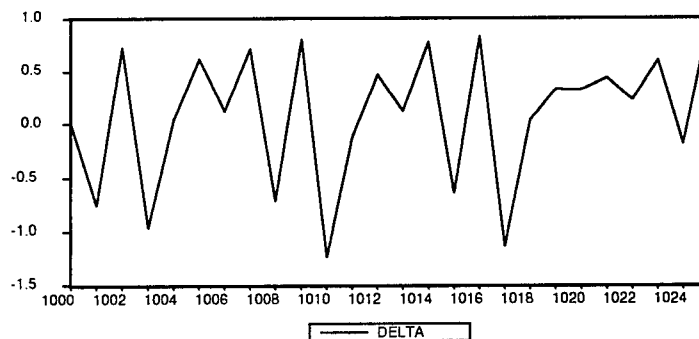


Figure 4.16 Divergence of forecast

this, the forecast value differs from the actual by thirty percent of the range in the first iteration.

This result confirms that the data is more effectively forecast using a deterministic chaotic model.

Rossler equations

The first two examples used discrete maps to generate the series of data that were analyzed. This example uses a system of differential equations. The system is known as the Rossler equations and is given by:

$$\begin{array}{ll} x'(t) = -(y(t) + z(t)) & x(0) = -1 \\ y'(t) = x(t) + ay(t) & y(0) = 0 \\ z'(t) = b + x(t)z(t) - cz(t) & z(0) = 0 \end{array}$$

The parameter values are $a = 0.2$, $b = 0.2$, and $c = 5.7$. These are "classic" parameter values and are known to lead to chaotic motion. To build a time series, a fourth order Runge-Kutta method is used to estimate the value of the system in time using a time step of 0.001 for the numerical approximation. The goal here is to get a set of one thousand data points which reasonably represents the attractor. Using only

the first thousand points with such a small step size only shows a small portion of the attractor. Increasing the step size is not a good solution since accuracy in the numerical method is sacrificed. The method used here, which preserves accuracy in the numerical algorithm, is to sample the points generated by the numerical method at a time step of 0.15. This value is chosen based on experimentation with various values and seems to create a time series which is representative of points on the attractor.

Since the method described in this thesis focuses on a scalar valued time series, only the x -values are analyzed. The choice of this series is arbitrary and the description of analysis which follows is believed to be typical of the results when applying the method to the y and z -values as well.

To obtain a better representation of the attractor, the data set starts at $t = 15$. This omits any early transient type behavior. To accomplish this the method described above was used to generate 2,000 data points and the first set of 1,000 discarded.

The time plot for the generated data set is shown in Figure 4.17 below. This plot shows evidence of

more structure than was observed in the earlier examples. It seems clear there is probably a cycle but no stable period length or amplitude seem obvious.

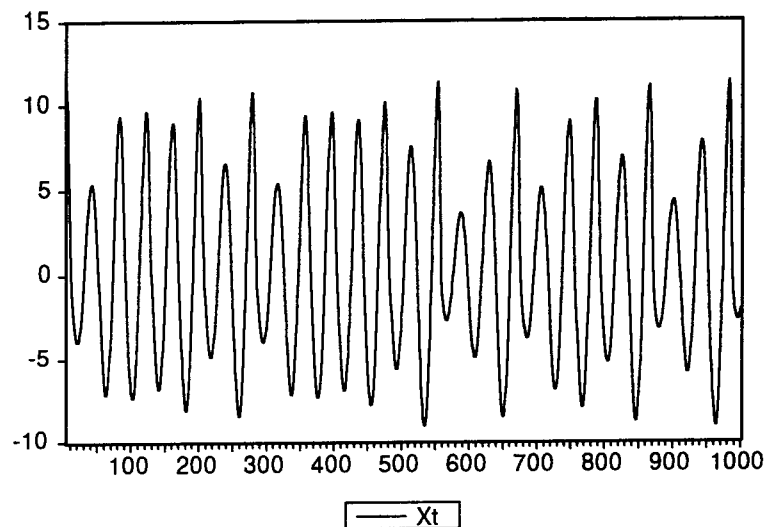


Figure 4.17 Time plot

Figure 4.18 shows the phase portrait for the Rossler x-values using a lag period of one (this corresponds to a time lag of $t = 0.15$). Here, as the points are plotted in phase space, they are connected to the previous point with a straight line. This representation is then an approximation of the path followed by a point in phase space.

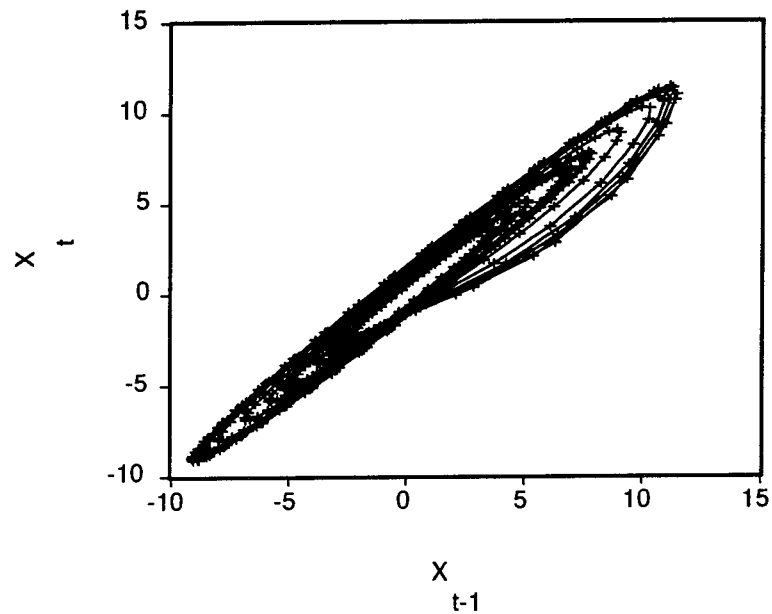


Figure 4.18 Phase portrait

Using this method, it appears the path followed is roughly elliptical and this suggests use of the elliptical model described in Chapter 2. Before a model is built, the diagnostic tools are applied in search of further support for the deterministic hypothesis.

The summary of results from diagnostic tools is given in Table 4.17 below. The estimated Hurst coefficient is larger than 1.0. Based on the results

observed for the strictly increasing sequence in Chapter 2, however, this estimate is believed to be larger than the actual value and the true value is probably close to, but less than 1.0.

Hurst coefficient	1.04
Hurst coefficient for scrambled data set	0.563
Correlation dimension	0.95
Largest Lyapunov exponent	0.203

Table 4.17 Summary of Diagnostic tools

The value of H confirms the presence of persistent trends which is apparent in the time plot. Another interesting aspect of using R/S analysis is that, according to Peters, it can approximate the cycle length of a system in some cases. In Figure 4.19, the values of $\log(R/S)_n$ are plotted against $\log(n)$. This plot is approximately linear with slope 1.04 for values of n less than fifty. After this, the slope is close to zero. This indicates that the approximate cycle length is fifty and this seems in agreement with the cycles seen in the time plot.

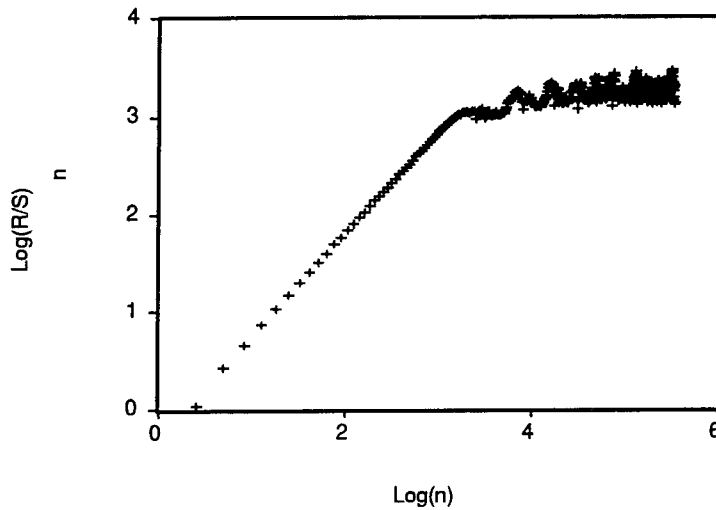


Figure 4.19 R/S plot

The Hurst coefficient after shuffling is very close to the expected H for a random process and therefore supports the deterministic hypothesis. The value of the correlation dimension shows it may be possible to model the system using only one variable. Finally, the low value of the correlation dimension and the positive largest Lyapunov exponent show the data may be the result of a low dimensional chaotic system and this is support for the deterministic hypothesis.

The autocorrelations are shown in Table 4.18 and show strongest dependence on the first and second lag









































































Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob	
		1	0.984	0.984	487.81	0.000
		2	0.938	-0.943	931.72	0.000
		3	0.864	0.009	1309.8	0.000
		4	0.769	0.073	1609.3	0.000
		5	0.654	-0.089	1826.7	0.000
		6	0.525	-0.141	1967.2	0.000
		7	0.386	-0.106	2043.2	0.000
		8	0.240	-0.074	2072.7	0.000
		9	0.092	-0.068	2077.0	0.000
		10	-0.056	-0.069	2078.6	0.000
		11	-0.199	-0.065	2099.0	0.000
		12	-0.334	-0.054	2156.6	0.000
		13	-0.458	-0.038	2265.1	0.000
		14	-0.568	-0.026	2432.1	0.000
		15	-0.661	-0.049	2658.8	0.000
		16	-0.736	-0.100	2940.2	0.000
		17	-0.791	-0.076	3265.9	0.000
		18	-0.825	-0.013	3621.3	0.000
		19	-0.839	-0.012	3989.2	0.000
		20	-0.832	-0.039	4351.6	0.000
		21	-0.804	-0.039	4691.2	0.000
		22	-0.758	-0.023	4993.2	0.000
		23	-0.693	-0.014	5246.5	0.000
		24	-0.613	-0.014	5444.9	0.000
		25	-0.519	-0.014	5587.2	0.000
		26	-0.413	-0.011	5677.8	0.000
		27	-0.299	-0.005	5725.3	0.000
		28	-0.179	-0.000	5742.5	0.000
		29	-0.057	-0.002	5744.2	0.000
		30	0.065	-0.009	5746.5	0.000
		31	0.183	-0.009	5764.5	0.000
		32	0.295	-0.004	5811.4	0.000
		33	0.399	-0.005	5897.0	0.000
		34	0.490	-0.015	6026.8	0.000
		35	0.568	-0.025	6201.5	0.000
		36	0.631	-0.030	6417.1	0.000

Table 4.18 Autocorrelations

periods. The autocorrelations after shuffling are shown in Table 4.19 and show that the structure previously observed no longer exists. Both of these results support the deterministic hypothesis and the choice of the first lag variable for use in the initial regression model.

Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob
		1 0.006	0.006	0.0424	0.837
		2 -0.011	-0.011	0.1557	0.925
		3 0.011	0.012	0.2856	0.963
		4 0.001	0.001	0.2878	0.991
		5 -0.014	-0.014	0.4867	0.993
		6 0.030	0.030	1.3935	0.966
		7 -0.007	-0.008	1.4462	0.984
		8 0.046	0.047	3.5681	0.894
		9 0.020	0.019	3.9871	0.912
		10 0.022	0.023	4.4682	0.924
		11 -0.011	-0.011	4.5903	0.949
		12 0.046	0.046	6.7820	0.872
		13 0.002	0.002	6.7858	0.913
		14 -0.018	-0.019	7.1020	0.931

Table 4.19 Autocorrelations for shuffled data

Given the elliptical shape of the phase portrait and the strongest dependence on the first lag variable, the initial regression model used is the one

developed in Chapter 3 from the standard equation of an ellipse:

$$x_t^2 = \beta_0 + \beta_1 \cdot x_t + \beta_2 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-1} + \beta_4 \cdot x_t \cdot x_{t-1}$$

Using this model, the coefficients found correspond to a hyperbola, rather than to an ellipse. This is determined by returning to the standard equation for an ellipse and evaluating $B^2 - 4 \cdot A \cdot C$. For an ellipse, this value is negative. In this analysis, the value for the resulting coefficients is small but positive.

To look for an explanation for this unexpected result, a phase portrait plotting x_t^2 against x_{t-1} is generated again using the technique of connecting sequential points. This plot (Figure 4.20) is similar in nature to that for the Henon equations. The key difference is that while the apparent axis of symmetry for the Henon phase portrait was vertical, here the apparent axis is not. This indicates a similar model to that used for the Henon equations may be appropriate but a "mixed term", that is $x_t \cdot x_{t-1}$, is required to account for the changed axis of symmetry.

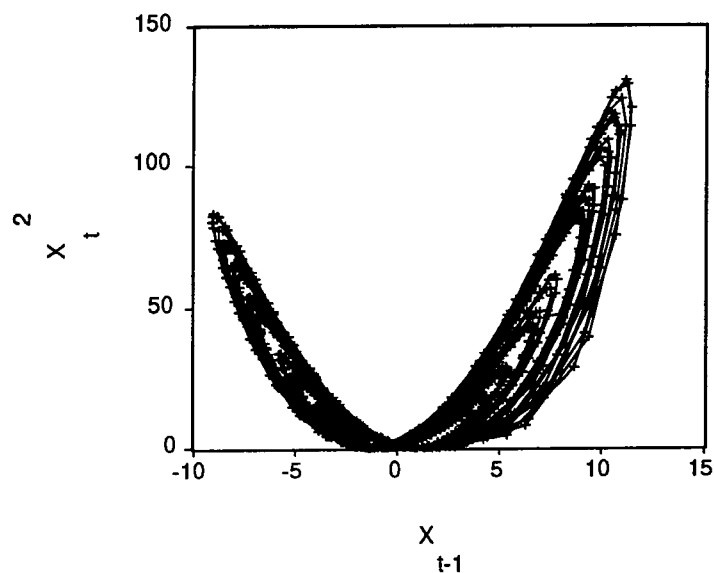


Figure 4.20 Phase portrait for x_t^2

Based on these results, the next regression model used is:

$$x_t^2 = \beta_0 + \beta_1 \cdot x_{t-1}^2 + \beta_2 \cdot x_t \cdot x_{t-1} + \beta_3 \cdot x_{t-1}$$

The result of this regression (Table 4.20) is much better than for the first model. The next step is to look at the autocorrelations of the residual series to determine whether further attempts at improving the model are needed. The autocorrelations for the residual series are shown in Table 4.21.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
$(X_{t-1})^2$	-0.863099	0.006424	-134.3543	0.0000
$X_t * X_{t-1}$	1.853676	0.006475	286.2768	0.0000
X_{t-1}	0.067165	0.005995	11.20272	0.0000
C	1.031850	0.043558	23.68894	0.0000
R-squared	0.998873	Mean dependent var	28.21846	
Adjusted R-squared	0.998869	S.D. dependent var	28.37966	
S.E. of regression	0.954343	Akaike info criterion	-0.089457	
Sum squared resid	903.4837	Schwartz criterion	-0.069763	
Log likelihood	-1364.713	F-statistic	292966.4	
Durbin-Watson stat	0.323941	Prob[F-statistic]	0.000000	

Table 4.20 Regression for second model





























Autocorrelation	Partial Correlation	AC	PAC	Q-Stat	Prob	
		1	0.838	0.838	700.88	0.000
		2	0.526	-0.588	977.56	0.000
		3	0.309	0.522	1073.1	0.000
		4	0.223	-0.317	1123.0	0.000
		5	0.199	0.281	1162.6	0.000
		6	0.184	-0.165	1196.7	0.000
		7	0.165	0.137	1224.1	0.000
		8	0.141	-0.087	1244.1	0.000
		9	0.116	0.067	1257.6	0.000
		10	0.094	-0.043	1266.4	0.000
		11	0.077	0.040	1272.4	0.000
		12	0.068	-0.006	1277.0	0.000
		13	0.069	0.049	1281.8	0.000
		14	0.080	0.010	1288.3	0.000

Table 4.21 Autocorrelations for residuals

The autocorrelations show evidence of remaining structure. After the first lag period, the strongest correlation is in the second lag variable. This indicates the next step is to look at a plot of the residuals against the x-values lagged two periods. This plot is given in Figure 4.21.

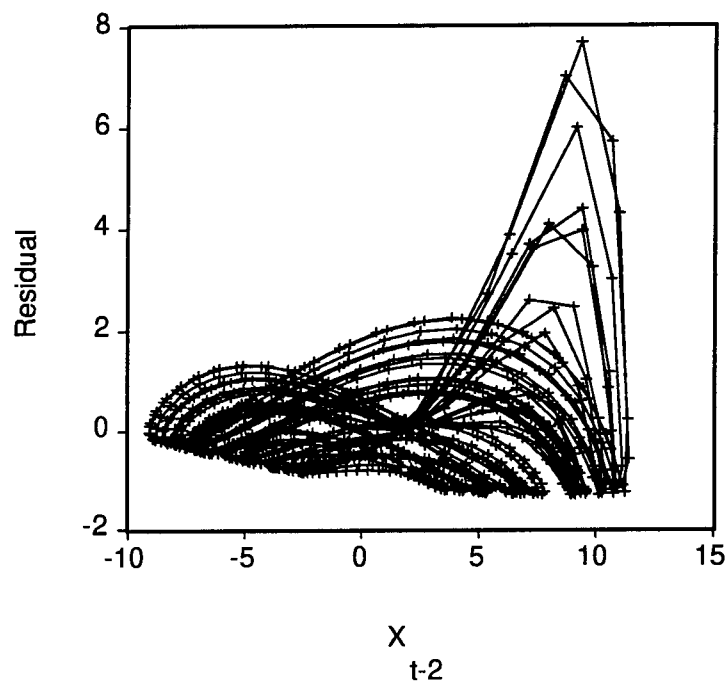


Figure 4.21 Plot of residuals vs. x_t lagged two periods

The plot confirms the presence of structure but no obvious geometric model is suggested. The most appropriate model suggested is again a parabola and based on this the revised regression model is:

$$x_t^2 = \beta_0 + \beta_1 \cdot x_{t-1}^2 + \beta_2 \cdot x_t \cdot x_{t-1} + \beta_3 \cdot x_{t-2}^2 + \beta_4 \cdot x_t \cdot x_{t-2} + \beta_5 \cdot x_{t-1} + \beta_6 \cdot x_{t-2}$$

The results of the regression using this model are shown in Table 4.22. For this model, the R-squared value is very close to 1.0 and therefore no further improvements are attempted.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
Xt*Xt1	2.410422	0.006883	350.1933	0.0000
Xt*Xt2	-0.760266	0.007143	-106.4405	0.0000
(Xt1)^2	-0.875962	0.010673	-82.07267	0.0000
(Xt2)^2	0.215751	0.003579	60.27493	0.0000
Xt1	0.080216	0.009425	8.510708	0.0000
Xt2	-0.091101	0.009478	-9.611700	0.0000
C	0.038175	0.014250	2.678922	0.0075
R-squared	0.999926	Mean dependent var	28.21846	
Adjusted R-squared	0.999926	S.D. dependent var	28.37966	
S.E. of regression	0.244540	Akaike info criterion	-2.809754	
Sum squared resid	59.14177	Schwartz criterion	-2.775289	
Log likelihood	-7.005493	F-statistic	2233347.	
Durbin-Watson stat	1.137811	Prob[F-statistic]	0.000000	

Table 4.22 Regression for second model

The model used is for the series values squared. The next step, then, is to solve the model for x_t in terms of the other lags so it can be used to forecast the data. This is done with the use of the quadratic formula as described in Chapter 3. First, the model is rewritten as a quadratic in x_t :

$$x_t^2 - (\beta_2 \cdot x_{t-1} + \beta_4 \cdot x_{t-2}) \cdot x_t - (\beta_0 + \beta_1 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-2}^2 + \beta_5 \cdot x_{t-1} + \beta_6 \cdot x_{t-2}) = 0$$

Then, applying the quadratic formula:

$$x_t = (1/2 \cdot a) \cdot [-b \pm ((b)^2 - 4 \cdot a \cdot c)^{0.5}]$$

with $a = 1$

$$b = -(\beta_2 \cdot x_{t-1} + \beta_4 \cdot x_{t-2})$$

$$c = -(\beta_0 + \beta_1 \cdot x_{t-1}^2 + \beta_3 \cdot x_{t-2}^2 + \beta_5 \cdot x_{t-1} + \beta_6 \cdot x_{t-2})$$

The result is an expression for x_t in terms of only lag variables. The best answer for which root to use is essentially a guess based on patterns in the data set.

Looking at the time plot, there are intervals in which the trajectory is concave up and intervals in

which it is concave down. Using the positive root in the forecasting equation above results in a concave down path in time space while the negative root yields a concave up path. Therefore, the initial sign for the root is chosen based on whether the system appears to be in a concave up or concave down interval at the point the desired forecast begins. The exact points defining the ends of these "intervals of concavity" are not easily determined. The actual change in concavity occurs when the change in the slope reverses sign. The difference between the present forecast value and the previous forecast value is used to approximate the slope of the forecast "function". The difference between the current slope and the previous slope is used to approximate the change in slope. The forecast is built, then, by initially choosing the sign of the root corresponding to the apparent concavity at the beginning of the forecast. With the two values used to generate the forecast and the forecast itself, an initial change in slope is computed and this will be positive or negative. When the change in slope is positive (negative) the sign of the root used in the forecast equation remains the

same until the change in slope becomes negative (positive) at which point the opposite sign of the root is used. This sign will be used until the change in slope becomes positive (negative) again and so on. The resulting forecast is shown in Figure 4.22 below.

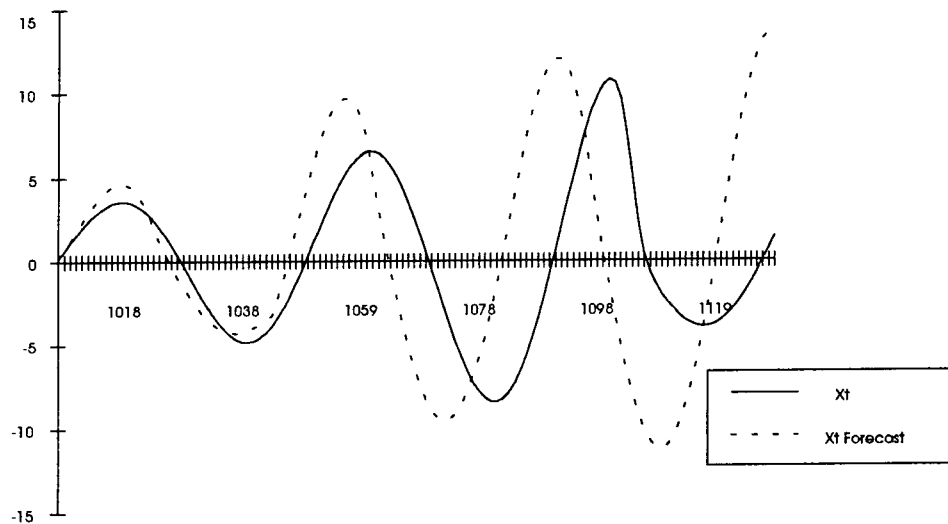


Figure 4.22 Deterministic forecast

The difference between the actual and the forecast values is plotted in Figure 4.23 below and shows that the deterministic model is able to forecast approximately fifty iterations before differing by more than 1.5 which is roughly ten percent of the

maximum value of the series. This shows that it is reasonably accurate for one full "period".

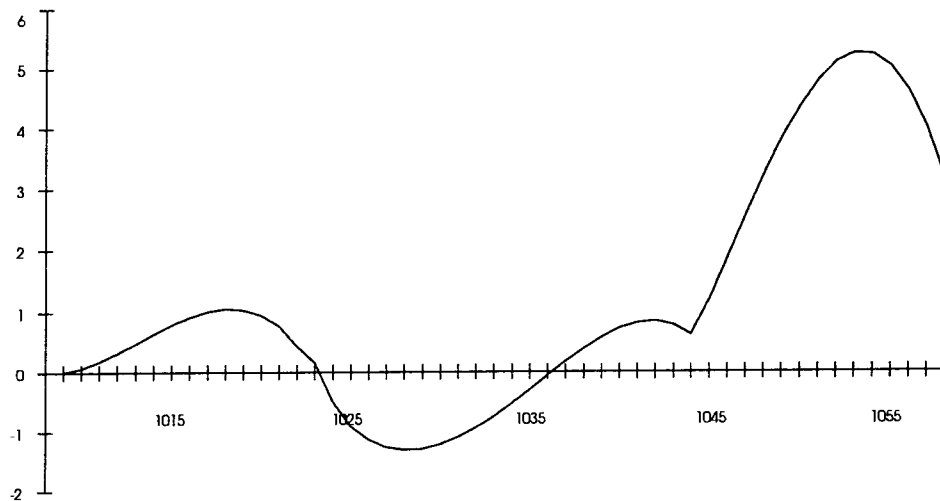


Figure 4.23 Divergence for deterministic model

Table 4.23 shows the result of building the ARIMA model for the data. The fit for this model is much better than the ARIMA models for the previous functions but the R-squared value is still lower than that for the deterministic model. The result of forecasting using this model is shown in Figure 4.24. The model is very good for the first ten iterations but forecast values quickly diverge from the actual values after this.

Variable	Coefficient	Std. Error	T-Statistic	Prob.
C	0.185443	0.173340	1.069824	0.2850
AR(1)	3.231596	0.020687	156.2123	0.0000
AR(2)	-4.250126	0.057046	-74.50377	0.0000
AR(3)	2.762021	0.057012	48.44627	0.0000
AR(4)	-0.757413	0.020648	-36.68260	0.0000
R-squared	0.999795	Mean dependent var	0.191798	
Adjusted R-squared	0.999794	S.D. dependent var	5.311309	
S.E. of regression	0.076157	Akaike info criterion	-5.144906	
Sum squared resid	5.747710	Schwartz criterion	-5.120288	
Log likelihood	1153.900	F-statistic	1209640.	
Durbin-Watson stat	1.279234	Prob(F-statistic)	0.000000	

Table 4.23 Regression for ARIMA model

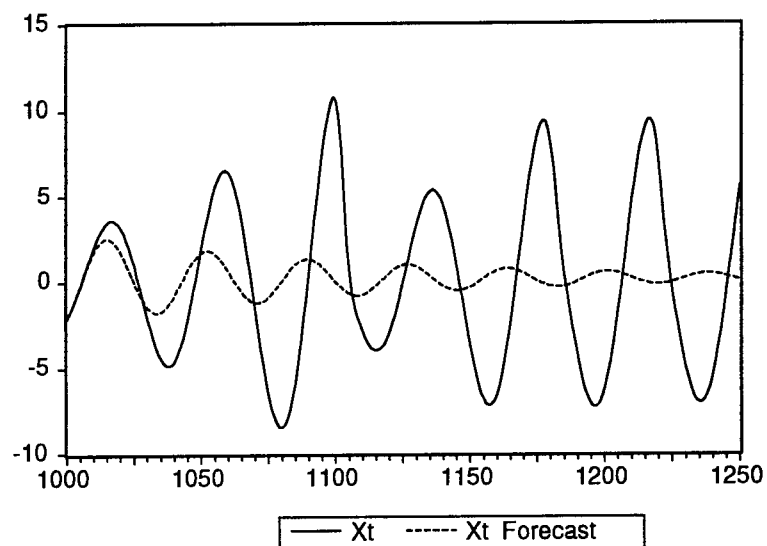


Figure 4.24 Forecast and actual values for ARIMA

The difference between the actual and forecast values is plotted in Figure 4.25 and shows the divergence is greater than 1.5 after approximately twenty iterations and more than twice that by the fortieth iteration.

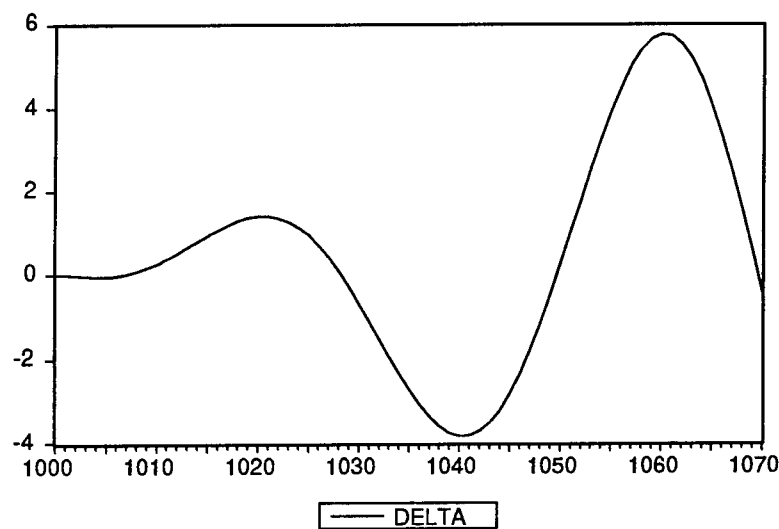


Figure 4.25 Divergence for ARIMA model

This shows that again the deterministic model is more effective in forecasting the data.

CHAPTER FIVE

SUMMARY

Overview

This chapter consists of four sections. The first section, results, is a brief summary of the work done in this study. The second section contains conclusions drawn in light of the results presented in the first section. The third section is titled recommendations and gives areas in which further study would likely lead to improvements on the methods presented in this thesis. Extensions, the final section, briefly presents additional areas where the methods presented here should be applied and the effectiveness evaluated.

Results

The analysis presented in Chapter 4 began with a data generated from a very simple chaotic system, the logistic equation, which is a one dimensional system. The diagnostic tools effectively revealed the presence of deterministic chaos. Using these tools with the method described in Chapter 3, the generating function was completely recovered in the model equation. This result was far superior to that achieved with the

traditional ARIMA model. Additionally, the analysis was consistent for both parameter values.

With success achieved in a simple case, the analysis was then applied to data from a more complicated set of equations, the Henon equations, which come from a two dimensional system. Again the diagnostic tools accurately identified the presence of deterministic chaos. The model was not as easily built as was the case for the logistic equation. Following the steps outlined, however, still resulted in the complete recovery of the generating functions. As with the logistic equation, the deterministic model was far superior to the ARIMA model.

To further test the method, data generated from a three dimensional set of differential equations, the Rossler equations, was studied. The diagnostic tools continued to be effective but the modeling process was significantly more complicated. Since the generating functions were differential equations, modeling the system with a discrete map did not reproduce the generating equations. The discrete model was, however, still very effective in forecasting the data. Additionally, the deterministic forecast was more

effective in forecasting the data than the ARIMA model.

The previous results all support the goal of the thesis, to demonstrate the effectiveness of a chaotic model for random looking data sets. Another result which deserves mention is the R/S plot for the strictly alternating series examined in Chapter 3. The plot showed two essentially parallel lines. A similar plot to this, not included in this thesis, was observed when using R/S analysis on the logistic equation for the parameter $\lambda = 3.6$.

Conclusions

The results here demonstrate that it is possible to recover, or at least effectively approximate, the functions which generated a series of random looking data. This means that a system which has a low dimensional deterministic mechanism can be effectively modeled in the same way using only a series of experimental observations. Experimental results which appear to be random noise, then, should be tested using the tools described in this thesis to test whether the random looking results may actually have a very simple deterministic mechanism.

Recommendations

The appearance of two distinct lines in the R/S plot for the alternating series is not fully understood. This may present a challenge when studying data sets with a high degree of antipersistence. Further study of data sets with small Hurst coefficients is suggested to fully understand this result.

In applying R/S analysis to pseudo-random time series, the need for some distribution theory has become apparent. For small sample sizes such as those used in this study, the expected Hurst coefficient for a random process appears to be significantly different from 0.5. A theoretical development of the expected coefficient and a confidence interval for rejecting the deterministic hypothesis is recommended. An alternative to this approach would be the development of an approximate distribution through experimentation using a large number of independent series of pseudo-random numbers may also be useful.

Another aspect of the Hurst coefficient which deserves study is the constant term in the linear regression use to approximate H . This is simply the

intercept term for the log/log plot of R/S against n . It appears that this term should have some significance and may be very useful for small data sets. More study is required to determine if this coefficient does have any significance and, if so, whether it would be useful in analysis such as that described here.

Extensions

The techniques demonstrated using data from known chaotic functions should next be applied to random looking data from real world systems. Preliminary attempts on data sets from the Dow Jones average, monthly rainfall totals, and sunspot numbers revealed some degree of structure in all cases but no success was attained in modeling the systems. It is likely that these systems are too complicated for the techniques described here. Further research should focus on data sets which have simpler mechanisms than those described above. An example might be data taken from a chaotic water wheel such as the one described in [17, p.27]. Briefly, this water wheel has a flow of water into the cups at the top and each cup has a drain hole in the bottom. For some water flow rates

and hole sizes, the motion of the water wheel becomes chaotic. This behavior is similar to that of the logistic equation which is chaotic for some parameter values. The water wheel may be a good candidate for study since the physical laws governing its motion are relatively simple. Another possible approach is to study data sets generated from known chaotic functions which are more complex than those studied in this work.

Hopefully, by increasing the complexity of the systems in small increments, new methods will evolve which ultimately will allow the researcher to deal with complicated systems such as the economy and the weather.

APPENDIX A

HURST COEFFICIENT PROGRAM

Description

The following BASIC program was written by Dr. William Lesso and has not been modified from its original form. The program requires that input files be in Lotus format, that is the first entry in the data set must be the number of points contained in the data set. Additionally, the files must have a .prn extension. The output file has five columns of numbers and from left to right these are: (1) n , (2) $(R/S)_n$, (3) $\log(n)$, (4) $\log(R/S)_n$, and an estimate of H given by $(\log(R/S)_n)/\log(n)$.

```
5 KEY OFF
10 REM CALCULATION OF HURST COEFFICIENT
20 ID = 0
24 N = 2500
25 DIM X(N)
100 TV$ = "UNIV OF TEXAS"
110 CV$ = "CHAOS ANALYSIS PROJECT"
115 PV$ = "HURST COEFFICIENT PROGRAM"
130 DIM CL$(10)
135 NL = 8
140 CL$(1) = "      N      NEW PROBLEM"
150 CL$(2) = "      D      DISK READ/WRITE"
160 CL$(3) = "      L      LIST DATA"
170 CL$(4) = "      C      CHANGE DATA"
180 CL$(5) = "      S      SOLVE"
190 CL$(6) = "      P      PRINT RESULTS"
```



```

200 CL$(7) = "      E      EXAMPLE PROBLEM"
210 CL$(8) = "      Q      QUIT"
250 CLS
251 PRINT TAB(25); , TV$
252 PRINT : PRINT TAB(25); , CV$
253 PRINT : PRINT TAB(25); , PV$
300 CLS
301 PRINT TAB(25); , TV$
302 PRINT : PRINT TAB(25); , CV$
303 PRINT : PRINT TAB(25); , PV$
304 PRINT TAB(29); DV$
310 FOR I = 1 TO NL: PRINT : PRINT TAB(25); CL$(I):
NEXT I
320 PRINT : PRINT TAB(30); : INPUT "COMMAND "; C$
325 I = 0
330 IF C$ = "" THEN 330
331 IF C$ = "N" THEN I = 1
332 IF C$ = "D" THEN I = 2
333 IF C$ = "L" THEN I = 3
334 IF C$ = "C" THEN I = 4
335 IF C$ = "S" THEN I = 5
336 IF C$ = "P" THEN I = 6
337 IF C$ = "E" THEN I = 7
338 IF C$ = "Q" THEN I = 8
340 IF I = 0 THEN 300
350 PRINT
351 PRINT : PRINT TAB(25); CL$(I)
352 PRINT TAB(20); "      CONFIRM ";
360 C$ = INKEY$: IF C$ = "" THEN 360 ELSE IF ASC(C$)
<> 13 AND C$ <> "Y" THEN PRINT CHR$(29); CHR$(30);
CHR$(29); : GOTO 320
370 IF C$ = "X" THEN 300
400 CLS : ON I GOSUB 1000, 2000, 3000, 4000, 5000,
6000, 7000, 8000
410 GOTO 300
1000 REM NEW PROBLEM
1005 CLS : PRINT : PRINT TAB(20); "CURVE FITTING
PROGRAM, 1.0"
1010 PRINT : PRINT TAB(20); "NEW PROBLEM"
1015 ID = 1
1019 PRINT
1020 PRINT : PRINT TAB(20); "ENTER THE NUMBER OF DATA
PAIRS ": INPUT N
1027 PRINT : PRINT TAB(20); "ENTER THE DATA      "

```

```

1030 FOR I = 1 TO N: INPUT X(I), Y(I): NEXT I
1950 FOR IT = 1 TO 1000: NEXT
1990 RETURN
2000 REM DISC READ/WRITE
2010 CLS : PRINT : PRINT TAB(20); "DISC READ/WRITE"
2030 PRINT : PRINT TAB(20); "CURRENT MAX-LIKE HD FILES
ON DISK ARE:": PRINT
2040 PRINT TAB(20); : FILES "*.PRN"
2050 PRINT : PRINT TAB(20); "R - READ AN EXISTING FILE
FROM DISK"
2060 PRINT : PRINT TAB(20); "W - WRITE CURRENT FILE TO
DISK"
2070 PRINT : PRINT TAB(20); "X - DO NOTHING, RETURN TO
MAIN MENU"
2080 PRINT : PRINT TAB(20); "YOUR CHOICE (R,W,X)"; :
INPUT C$
2090 IF C$ = "X" THEN RETURN ELSE IF C$ = "W" THEN
2300 ELSE IF C$ <> "R" THEN PRINT TAB(20); "PLEASE
ENTER R, W OR X ! ! !": GOTO 2050
2100 REM READ DATA FROM DISK
2110 F$ = "": PRINT : PRINT TAB(20); "FILE NAME TO BE
READ - (.PRN WILL BE ADDED)"; : INPUT F$
2120 IF F$ = "" THEN 2110
2130 F$ = F$ + ".PRN"
2135 ID = 1
2140 OPEN "I", #1, F$
2150 INPUT #1, N
2160 FOR I = 1 TO N
2170 INPUT #1, X(I)
2180 NEXT I
2200 CLOSE
2210 RETURN
2300 REM WRITE DATA TO DISK
2310 IF ID = 0 THEN CLS : GOTO 9000 ELSE F$ = "":
PRINT : PRINT TAB(20); : INPUT "FILE NAME FOR SAVING
(.PRN WILL BE ADDED) "; F$
2320 IF F$ = "" GOTO 2310
2340 F$ = F$ + ".PRN"
2350 OPEN "O", #1, F$
2360 PRINT #1, N, M
2370 FOR I = 1 TO N
2380 PRINT #1, X(I)
2390 NEXT I
2410 CLOSE

```

```

2420 RETURN
2950 FOR IT = 1 TO 1000: NEXT
2990 RETURN
3000 REM LIST DATA
3010 CLS : PRINT : PRINT TAB(20); "LIST DATA"
3015 IF ID = 0 THEN PRINT TAB(20); "NO DATA HAVE BEEN
ENTERED OR READ FROM DISK": FOR IT = 1 TO 1000: NEXT:
RETURN
3020 PRINT : PRINT TAB(20); "THE FOLLOWING DATA HAS
BEEN ENTERED":
3045 LN = 6
3046 PRINT : PRINT TAB(15); "      I          X(I)
Y(I)"
3050 FOR I = 1 TO N
3052 PRINT TAB(18); I, X(I)
3053 NEXT I
3055 IF LN >= 22 GOTO 9200
3065 LN = LN + 2: IF LN >= 30 THEN GOSUB 9200
3090 LN = LN + 2: IF LN >= 30 THEN GOSUB 9200
3400 PRINT : PRINT TAB(20); : INPUT "PRESS <ENTER> TO
RETURN TO MAIN MENU..."; C$
3990 RETURN
4000 REM CHANGE DATA
4010 CLS : PRINT : PRINT TAB(20); "CHANGE DATA"
4020 PRINT : PRINT TAB(20); "DO YOU WANT A LISTING OF
THE DATA - (Y/N) "; : INPUT C$
4030 IF C$ = "Y" OR C$ = "y" THEN GOSUB 3000
4920 FOR IT = 1 TO 1000: NEXT
4930 RETURN
5000 REM SOLVE
5010 CLS : PRINT : PRINT TAB(20); "SOLVE"
5011 IF ID = 0 THEN CLS : GOTO 9000 ELSE F$ = "":
PRINT : PRINT TAB(20); : INPUT "FILE NAME FOR SAVING
(.PRN WILL BE ADDED) "; F$
5012 IF F$ = "" GOTO 5011
5013 F$ = F$ + ".PRN"
5014 OPEN "O", #1, F$
5020 M = N / 2: NI = 3
5021 PRINT : PRINT TAB(20); "WORKING....."
5030 FOR I = 1 TO M
5035 LLO = 1: LHI = NI: RSUM = 0: CNT = 0
5040 SUM = 0: SUM2 = 0
5045 FOR J = LLO TO LHI
5050 SUM = SUM + X(J): SUM2 = SUM2 + X(J) * X(J)

```

```

5051 'PRINT J,X(J),SUM,SUM2
5055 NEXT J
5060 XBAR = SUM / NI
5068 'PRINT NI;LLO;LHI;SUM; SUM2;XBAR
5069 XMIN = 99999!: XMAX = -99999!: CUM = 0: SUMSD = 0
5070 FOR J = LLO TO LHI
5071 SUMSD = SUMSD + (X(J) - XBAR) ^ 2
5075 CUM = CUM + (X(J) - XBAR)
5080 IF CUM < XMIN THEN XMIN = CUM
5085 IF CUM > XMAX THEN XMAX = CUM
5090 NEXT J
5091 SD = (SUMSD / (NI - 1)) ^ .5
5095 RSUM = RSUM + (XMAX - XMIN) / SD: CNT = CNT + 1
5096 'PRINT J, RSUM, CNT
5100 LLO = LLO + NI: LHI = LHI + NI
5105 IF LHI <= N GOTO 5040
5110 RS = RSUM / CNT
5115 LNN = LOG(NI / 2)
5120 LNRS = LOG(RS)
5122 HURST = LNRS / LNN
5124 PRINT NI, RS, LNN, LNRS, HURST
5125 PRINT #1, NI, RS, LNN, LNRS, HURST
5130 NI = NI + 1
5500 NEXT I
5519 CLOSE
5700 INPUT "CONTINUE "; C$
5920 FOR IT = 1 TO 9000: NEXT
6000 REM PRINT RESULTS
6010 CLS : PRINT : PRINT TAB(20); "PRINT RESULTS"
6040 LN = 1
6046 PRINT : PRINT TAB(15); "      I      X(I)
Y(I)      Yhat(I)      %ERR"
6325 LN = LN + 2
6345 LN = LN + 2
6360 IF LN >= 30 THEN GOSUB 9200
6390 PRINT
6400 PRINT : PRINT TAB(20); : INPUT "PRESS <ENTER> TO
RETURN TO MAIN MENU..."; C$
6920 FOR IT = 1 TO 1000: NEXT
6930 RETURN
7000 REM EXAMPLE PROBLEM
7010 CLS : PRINT : PRINT TAB(20); "EXAMPLE PROBLEM"
7020 F$ = "SAMPLE": GOTO 2130
7025 GOSUB 3000

```

```
7030 RETURN
7920 FOR IT = 1 TO 1000: NEXT
7930 RETURN
8000 REM QUIT
8010 CLS : PRINT : PRINT TAB(20); "QUIT"
8020 FOR IT = 1 TO 1000: NEXT
8025 END
8030 RETURN
9000 IF ASC(C$) > 95 THEN C$ = CHR$(ASC(C$) - 32):
RETURN
9200 PRINT : PRINT TAB(20); " PRESS ENTER TO CONTINUE
"; : C$ = INKEY$
9205 LN = 0
9220 IF INKEY$ = "" THEN 9220 ELSE CLS : RETURN
```

APPENDIX B

CORRELATION INTEGRAL PROGRAM

Description

The following code is essentially a translation, into Turbo Pascal 5.0, of BASIC code recieved from Dr. William Lesso. The major contribution is a significant addition of documentation within the code.

```
{*****}
{ The algorithm is from similar BASIC code received from Dr. William }
{ Lesso. }
{ The code was created on 14 OCT 94 by J. Robert Bookhart, UT Austin }
{*****}
```

PROGRAM CorrelationIntegral;

```
{*****}
{ This program prompts the user for the names of the files to read data }
{ from and write data to, the number of points in the data file, the }
{ embedding dimension to use when reconstructing the attractor, the time }
{ lag used for reconstructing the phase space, the initial R value to use }
{ when counting the number of pairs of points separated by less than R and }
{ the increment used to increase R with each iteration. The output file }
{ file contains two columns of numbers. The first column is the value of }
{ the correlation integral for the R value and the second column is the R }
{ value. The only screen output is an update of the current iteration in }
{ progress and a message (with a beep) indicating the program is complete. }
{*****}
```

```

USES
  CRT;

```

```

TYPE

```

```

  Two_d_Vector = Array [1..1000,1..7] of Real;

```

```

  One_d_Vector = Array [1..1000] of Real;

```

```

  FileName = String[12];

```

```

{*****}
{ The two vector types allow a maximum data set length of 1,000 points }
{ and a maximum embedding dimension of 7. }
{*****}

```

```

VAR

```

```

  EmbeddingDimension      : Integer;      { Dimension for the }
                                   { reconstructed phase space }

```

```

  NumberOfPoints          : Integer;      { Number of points in data set }

```

```

  TimeLag                 : Integer;      { Lag time used to }
                                   { reconstruct the attractor }

```

```

  R_Value                 : Real;          { Initial value of the distance }
                                   { used when counting the }
                                   { number of pairs of points }
                                   { within distance R }

```

```

  R_Increment             : Real;          { step size to increase the value }
                                   { of R with each iteration }

```

```

  Z                       : Two_d_Vector; { The points on the }
                                   { reconstructed attractor }

```

```

  X                       : One_d_Vector; { The original series values }
                                   { read in from a data file }

```

```

  OutputFile,InputFile    : Text;
  InputFileName           : FileName;     { File name to read data from }

```

```

  OutputFileName          : FileName;     { File name to write data to }

```

```

Theta          : LongInt;    { Used to store the count of the      }
                                   { number of pairs of points      }
                                   { separated by less than R      }

```

```

PROCEDURE GetInputValues (VAR Val1, Val2, Val3      : Integer;
                          VAR Val4, Val5           : Real;
                          VAR Name1, Name2         : FileName);

```

```

{*****}
{ This procedure prompts the user to input the read file name, the      }
{ desired output file name, the number of points in the read file, the  }
{ embedding dimension to use to reconstruct the attractor, the time lag  }
{ to use when reconstructing the attractor, the initial R value and the }
{ the value with which to increment R with each iteration. These       }
{ inputs are then passed to the main program.                           }
{*****}

```

```

BEGIN (* GetInputValues *)

```

```

    Clrscr;
    Writeln ('Enter the name of the file to be read in:');
    Readln (Name1);
    Writeln ('Enter the name of the file to output data to:');
    Readln (Name2);
    Writeln ('Input the Number of points in the data set:');
    Readln (Val1);
    Writeln ('Input the Embedding dimension to be used:');
    Readln (Val2);
    Writeln ('Input the time lag used to construct the phase space:');
    Readln (Val3);
    Writeln ('Input the initial R value:');
    Readln (Val4);
    Writeln ('Input the value to increment R with each iteration:');
    Readln (Val5);
End; (* GetInput Values *)

```



```
PROCEDURE ReadFile (VAR DataList : One_d_Vector);
```

```
{*****}  
{ This procedure reads the data from a file into memory as a one }  
{ dimensional array called DataList }  
{*****}
```

```
VAR
```

```
  I : Integer;           { Looping variable }
```

```
BEGIN (* ReadFile *)
```

```
  Assign (InputFile , InputFileName );
```

```
  Reset (InputFile);
```

```
  For I := 1 to NumberOfPoints DO
```

```
    Readln(InputFile,DataList[I]);
```

```
  Close(InputFile)
```

```
END; (* ReadFile *)
```

PROCEDURE ReconstructPhaseSpace;

```
{*****}
{ This procedure reconstructs the attractor in phase space with dimension }
{ equal to the embedding dimension, n. The points on the attractor are }
{ ordered n-tuples. That is, a point, z, on the attractor is of the form }
{  $z_t = (x_t, x_{t-1}, x_{t-2}, \dots, x_{t-(n-1)})$ . The variable Z[I,J] is used to }
{ represent these n-tuples. The index, I, represents the number of the }
{ point on the attractor and the index, J, represents the Jth component }
{ of that n-tuple. For example, using the point  $z_t$  above, Z[t,2] holds }
{ the value  $x_{t-1}$ . }
{*****}
```

VAR

I,J : Integer; { counter variables used in loops}

BEGIN (* ReconstructPhaseSpace *)

FOR I := 1 to NumberOfPoints DO

FOR J := 1 to EmbeddingDimension DO

Z[I,J] := X[I + (J-1) * TimeLag];

END; (* ReconstructPhaseSpace *)

PROCEDURE CountPoints (VAR NumberOfPoints : Integer);

VAR

DistanceBetweenPoints : Real; { distance between pairs of }
 { points on the attractor }

Theta2 : Integer; { has the value 1 if the pair of }
 { points is separated by less }
 { than R, 0 otherwise }

J,K,L : Integer; { counters for loops }

Lag : Integer; { used as an index when }
 { finding the distance }
 { between pairs of points }

BEGIN (* CountPoints *)

Lag := 1;

Theta := 0;

Theta2 := 0;

FOR K := 1 to NumberOfPoints DO

BEGIN (* For Loop K *)

FOR L := 1 to NumberOfPoints DO

BEGIN (* For Loop L *)

DistanceBetweenPoints := 0;

FOR J := 1 to EmbeddingDimension DO

DistanceBetweenPoints := DistanceBetweenPoints
 + Sqr(Z[Lag,J] - Z[L,J]);

DistanceBetweenPoints := Sqrt(DistanceBetweenPoints);

IF DistanceBetweenPoints > R_Value Then Theta2 := 0

ELSE Theta2 := 1;

Theta := Theta + Theta2;

END; (* For Loop L *)

Lag := Lag + 1;

END; (* For Loop K *)

END; (* CountPoints *)

```
PROCEDURE CalculateIntegral (NumberOfPoints: Integer);
```

```
{*****}
{ This procedure calculates the value of the correlation integral and }
{ writes it to the output file. }
{*****}
```

```
VAR
```

```
CorrelationInt : Real; { value of the correlation integral }
```

```
I : Integer; { looping variable }
```

```
BEGIN (* CalculateIntegral *)
```

```
Assign (OutputFile,OutputFileName);
```

```
Rewrite (OutputFile);
```

```
NumberOfPoints := NumberOfPoints - EmbeddingDimension * TimeLag;
```

```
FOR I := 1 to 13 DO
```

```
BEGIN (* For Loop *)
```

```
Writeln ('Doing the ',I,'th iteration');
```

```
CountPoints (NumberOfPoints);
```

```
CorrelationInt := Sqr(1/NumberOfPoints) * Theta;
```

```
Write (OutputFile,CorrelationInt:7:5);
```

```
Write (OutputFile, ' ');
```

```
Writeln (OutputFile,R_Value:7:5);
```

```
R_Value := R_Value + R_Increment;
```

```
CorrelationInt := 0;
```

```
END; (* For Loop *)
```

```
Close (OutputFile);
```

```
END; (* CalculateIntegral *)
```

BEGIN (* Main Program *)

```
  Clrscr;  
  GetInputValues (NumberOfPoints,EmbeddingDimension,TimeLag,R_Value,  
                  R_Increment,InputFileName,OutputFileName);  
  ReadFile (X);  
  ReconstructPhaseSpace;  
  CalculateIntegral (NumberOfPoints);  
  Sound(440);  
  Delay(2000);  
  NoSound;  
  Writeln ('WHEW, that was a lot of work...Hit <Enter> to go to DOS');  
  Readln;
```

End. (* Main Program *)•

APPENDIX C

LYAPUNOV EXPONENT PROGRAM

The following BASIC code is a revision of a program obtained from Dr. William Lesso. The code, in its original form, contained several significant errors in the algorithm. Dr. Lesso cites Edgar Peters as reference in the creation of this program and in researching the subject it appears the original source is a FORTRAN program provided in the paper [11]. This reference is recommended for further reading about the method used.

BASIC code

```
2 PI = 3.141592654#
5 KEY OFF:REM LARGEST LYAPUNOV EXPONENT
10 DIM X(1000),PT1(12),PT2(12)
20 DIM Z(1000,5)
30 OPEN "LYAP.PRN" FOR OUTPUT AS 2 LEN=500
40 VT$ = "###.#####   ### ##.####  ##.####"
60 PRINT"NPT,DIM,TAU,DT,SCALMX,SCALMN,EVOLV,LAG?"
70 INPUT NPT: REM NUMBER OF OBSERVATIONS
80 INPUT DIMEN: REM EMBEDDING DIMENSION
90 INPUT TAU: REM LAG TIME FOR PHAS SPACE
100 INPUT DT :REM TIME INCREMENT IN SERIES FOR
    NORMALIZING EXPONENT
110 INPUT SCALMX: REM MAXIMUM DIVERGENCE
120 INPUT SCALMN:REM MINIMUM DISTANCE
130 INPUT EVOLV: REM EVOLUTION TIME
140 IND = 1
150 INPUT LAG :PRINT"MINIMUM TIME BETWEEN PAIRS"
```

```

160 SUM = 0
170 ITS=0
180 OPEN "DELAY.PRN" FOR INPUT AS 1 LEN=2500:REM INPUT
    FILE
185 PRINT"READING DATA"
190 FOR I = 1 TO NPT
200     INPUT#1,X(I)
210 NEXT I
220 PRINT TAB(15) "DATA READ"
230 FOR I = 1 TO NPT-(DIMEN-1)*TAU
240     FOR J = 1 TO DIMEN
250         Z(I,J)=X(I+(J-1)*TAU)
260     NEXT J
270 NEXT I
275 PRINT TAB(15) "DATA FORMATTED"
280 NPT=NPT-DIMEN*TAU-EVOLV: REM MAX LENGTH OF PHASE
    SPACE
290 DI = 100000000#
300 FOR I = (LAG+1) TO NPT: REM FIND INITIAL PAIR
310     D = 0
320     FOR J = 1 TO DIMEN
330         D = D + (Z(IND,J)-Z(I,J))^2: REM CALCULATE
            DISTANCE
340     NEXT J
350     D = SQR(D)
360     IF (D>DI) OR (D< SCALMN) GOTO 390: REM STORE
        BEST POINT
370     DI = D
380     IND2=I
390 NEXT I
400 FOR J = 1 TO DIMEN:REM COORDINATES OF EVOLVED
    POINTS
410     PT1(J)=Z(IND+EVOLV,J)
420     PT2(J)=Z(IND2+EVOLV,J)
430 NEXT J
440 DF = 0
450 FOR J = 1 TO DIMEN: REM COMPUTE FINAL DIVERGENCE
460     DF = DF +(PT2(J)-PT1(J))^2
470 NEXT J
480 DF = SQR(DF)
490 ITS = ITS + 1
500 SUM = SUM +(LOG(DF/DI)/(EVOLV*DT*LOG(2)))
510 ZLYAP = SUM/ITS
520 PRINT #2, USING VT$; ZLYPA, EVOLV*ITS,DI,DF

```

```

540 INDOLD = IND2
550 ZMULT=1
560 ANGLMX=.3
570 THMIN = 3.14
575 REM LOOK FOR REPLACEMENT POINTS
580 FOR I = 1 TO NPT
590     III=ABS(INT(I-(IND+EVOLV)))
600 IF III<LAG GOTO 780
605 REM REJECT IF REPLACEMENT POINT IS TOO CLOSE TO
    ORIGINAL
610 DNEW = 0
620 FOR J = 1 TO DIMEN
630     DNEW = DNEW + (PT1(J) - Z(I,J))^2
640 NEXT J
650 DNEW = SQR(DNEW)
660 IF (DNEW>ZMULT*SCALMX) OR ( DNEW < SCALMN) GOTO
    780
670 DOT = 0
680 FOR J = 1 TO DIMEN
690     DOT = DOT + (PT1(J) - Z(I,J)) * (PT1(J) - PT2(J))
700 NEXT J
710 CTH=ABS(DOT/(DNEW*DF))
720 IF (CTH>1) THEN CTH=1
730 TH=PI/2 - ATN(CTH/SQR(1 - CTH^2)) :REM USE ARCTAN
    TO FIND ARCCOS(CTH)
740 IF (TH>THMIN) GOTO 780
750 THMIN=TH
760 DII=DNEW
770 IND2 = I
780 NEXT I
790 IF (THMIN<ANGLMX) GOTO 870
800 ZMULT=ZMULT+1
810 IF (ZMULT<5) GOTO 570
820 ZMULT = 1
830 ANGLMX=2*ANGLMX
840 IF (ANGLMX<3.14) GOTO 570
850 IND2=INDOLD+EVOLV
860 DII=DF
870 IND=IND+EVOLV
880 IF (IND>=NPT) GOTO 910
890 DI = DII
900 GOTO 400
910 END

```


BIBLIOGRAPHY

- [1] Makridakis, Spyros, Steven C. Wheelwright, Victor E. McGee. Forecasting: Methods and Applications. 2nd ed. New York: John Wiley & Sons, 1983.
- [2] Ott, Edward, Tim Sauer, James A. Yorke. Eds. Coping with Chaos: Analysis of Chaotic Data and the Exploitation of Chaotic Systems. New York: John Wiley & Sons, 1994.
- [3] Brock, William A. and Chera L. Sayers. "Is the Business Cycle Characterized by Deterministic Chaos?". Journal of Monetary Economics. 1988:71-90.
- [4] Harrell, Evans M. II. Course Notes Mathematics 6350, Georgia Institute of Technology Spring 1990.
- [5] Barnsley, Michael F. Fractals Everywhere. Boston: Academic Press, Inc., 1988.
- [6] Devaney, Robert L. An Introduction to Chaotic Dynamical Systems. 2nd ed. New York: Addison-Wesley, 1989.

- [7] Peters, Edgar E. Fractal Market Analysis: Applying Chaos Theory to Investment and Economics. New York: John Wiley & Sons, 1994.
- [8] Mandelbrot, Benoit B. The Fractal Geometry of Nature. New York: W.H. Freeman and Company, 1983.
- [9] Mandelbrot, Benoit B. and James R. Wallis.
"Robustness of the Rescaled Range R/S in the Measurement of Noncyclic Long Run Statistical Dependence". Water Resources Research Volume 5, Number 5:967-988.
- [10] Moon, Francis C. Chaotic Vibrations: An Introduction for Applied Scientists and Engineers. New York: John Wiley & Sons, 1987.
- [11] Wolf, Alan, Jack B. Swift, Harry L. Swinney, John A. Vastano. "Determining Lyapunov Exponents from a Time Series". Physica 16D 1985:285-317.
- [12] Brock, William A., W. Davis Dechert, Jose A. Scheinkman. A Test for Independence Based on the Correlation Dimension. Unpublished Essay, November, 1986.

- [13] Grassberger, Peter and Itamar Procaccia.
"Measuring the Strangeness of Strange
Attractors". Physica 9D 1983:189-208.
- [14] Lesso, William G. Analysis of the Wolfer
Sunspot Numbers as a Deterministic Nonlinear
System. Unpublished Essay.
- [15] Lilien, David M. Micro TSP. Ver. 5.0. Computer
software. Irvine, California: Quantitative
Micro Software, 1986.
- [16] Peters, Edgar E. Chaos and Order in the Capital
Markets. New York: John Wiley & Sons, 1991.
- [17] Gleick, James. Chaos: Making a New Science.
New York: Penguin, 1988.

VITA

James Robert Bookhart, Jr. was born in Sioux City, Iowa on July 12, 1969, the son of Neva Lorene Bookhart and James Robert Bookhart, Sr. After completing his work at East High School, Sioux City, Iowa, in 1987, he entered the United States Air Force Academy in Colorado Springs, Colorado. During the 1990 academic year he attended Iowa State University in Ames, Iowa. He received the degree of Bachelor of Science from the Air Force Academy in May, 1992. He then attended the Air Force's Specialized Undergraduate Pilot Training Program at Reese Air Force Base in Lubbock, Texas. He received his pilot's wings in July, 1993. In August, 1993, he entered The Graduate School at The University of Texas.

Permanent Address: 4312 Village Green Ct.
 Sioux City, Iowa 51106

This thesis was typed by the author.